

Francisco Spaeth

**MODELO PARA PREVISÃO DE VAZÃO FLUVIAL
BASEADO EM WAVELETS E REDES NEURAI**

Florianópolis
2008

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA
COMPUTAÇÃO**

Francisco Spaeth

**MODELO PARA PREVISÃO DE VAZÃO FLUVIAL
BASEADO EM WAVELETS E REDES NEURAIS**

Dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para obtenção do grau de Mestre em Ciências da Computação

Orientador: João Cândido Lima Dovicchi

Florianópolis, Março de 2008

MODELO PARA PREVISÃO DE VAZÃO FLUVIAL BASEADO EM WAVELETS E REDES NEURAI

Francisco Spaeth

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciências da Computação (Sistemas de Conhecimento) e aprovada na sua forma final pelo Programa de Pós-Graduação em Ciências da Computação.

Banca Examinadora

Mário Antônio Ribeiro Dantas, Dr
Coordenador do PPGCC

João Cândido Lima Dovicchi, Dr.
Orientador

Luiz Fernando Jacintho Maia, Dr.

Gilberto Arantes Carrijo, Ph.D.

Silvia Modesto Nassar, Dra.

“O sucesso é construído à noite.”

(Roberto Shinyashiki)

AGRADECIMENTOS

A todas as pessoas que diretamente ou indiretamente auxiliaram na realização deste trabalho.

À Universidade Federal e aos professores do programa pelo belo trabalho realizado no departamento de pós-graduação em Ciências da Computação.

A meus pais Udo e Marli por fornecerem palavras de motivação e entusiasmo para realização do mesmo, bem como auxílio financeiro durante parte do processo.

Ao orientador e amigo João Cândido Lima Dovicchi, pela criatividade inspiradora e orientação.

Aos professores, colegas e amigos Fernando Adrade Bastos e Juliano Tonizette Brignolli pelo voto de confiança ao enviar carta de referência para seleção do programa de pós-graduação em ciências da computação da Universidade Federal de Santa Catarina.

Ao meu irmão Felipe pela companhia no traslado.

À minha noiva, Adriana, pela ajuda nos momentos que antecediam minhas idas a Florianópolis.

Aos amigos que fiz e convivi parte do tempo enquanto no programa, Cláudio e Wemerson.

Aos integrantes do núcleo de informática da Universidade para o Desenvolvimento do Alto Vale do Itajaí de 2008, em especial Franciano, Jackson, Luiz, Marco, Marciel, Marcondes e Paulo.

*A meus pais Udo e Marli, meus irmãos
Frederico e Felipe e minha noiva Adriana.*

SUMÁRIO

1 - INTRODUÇÃO.....	13
1.1 - ORGANIZAÇÃO DO TRABALHO.....	15
2 - REFERENCIAL TEÓRICO.....	17
2.1 - CLIMATOLOGIA / METEOROLOGIA.....	17
2.1.1 - Complexidade do Clima.....	18
2.1.2 - Modelos de Previsão para Cheias.....	19
2.1.3 - Objeto para Validação do Modelo Proposto.....	19
2.2 - FOURIER.....	20
2.2.1 - Transformada de Fourier.....	21
2.2.2 - Transformada Inversa.....	22
2.2.3 - Transformada Discreta de Fourier.....	22
2.2.4 - Transformada Rápida de Fourier (FFT – Fast Fourier Transform).....	23
2.2.5 - Short Time Fourier Transform.....	24
2.2.6 - DSTFT – Discrete Short Time Fourier Transform.....	26
2.3 - WAVELETS.....	26
2.3.1 - Transformada Wavelet.....	28
2.3.2 - Transformada Inversa.....	29
2.3.3 - Análise Multiresolução (MRA – Multiresolution Analysis).....	29
2.3.4 - Transformada Discreta Wavelet (DWT – Discrete Wavelet Transform).....	32
2.3.5 - Famílias de Wavelet.....	32
2.3.6 - Aplicação de Wavelet.....	33
2.3.6.1 - Wavelets em Sinais Geofísicos.....	34
2.3.6.2 - Wavelets em Biomedicina.....	35
2.4 - REDES NEURAIIS ARTIFICIAIS (ANN – Artificial Neural Networks).....	37
2.4.1 - Fundamento Neurofisiológico.....	39
2.4.2 - Introdução a Redes Neurais Artificiais.....	41
2.4.3 - Topologia de Redes Neurais Artificiais.....	44
2.4.4 - Aprendizado em Redes Neurais.....	44
2.4.4.1 - Aprendizado Hebbiano.....	47
2.4.4.2 - Treinamento de Perceptrons.....	48
2.4.4.3 - Regra Delta.....	50
2.4.4.4 - Algoritmo de Retropropagação.....	51
2.4.5 - Redes MLP (Multilayer Perceptron).....	52
2.4.6 - Aproximação de Funções / Reconhecimento de Padrões.....	54
3 - MATERIAL E MÉTODO.....	56
3.1 - DESCRIÇÃO DOS ALGORITMOS.....	59
4 - RESULTADOS.....	62
4.1 - COMPARAÇÃO ENTRE WAVELETS.....	63
4.2 - COMPARAÇÃO JANELAMENTO / SOBREPOSIÇÃO.....	65
4.2.1 - Somente Rede Neural.....	66

4.2.2 - Chroma 6 – Análise de Multiresolução Nível 1.....	69
4.2.3 - Chroma 6 – Análise de Multiresolução Nível 2.....	72
4.2.4 - Chroma 6 – Análise de Multiresolução Nível 3.....	75
4.3 - COMPORTAMENTO DO MODELO EM RELAÇÃO AO CONJUNTO DE TREINAMENTO E ÉPOCAS DE TREINAMENTO.....	78
5 - DISCUSSÃO.....	81
6 - CONCLUSÕES.....	86
6.1 - TRABALHOS FUTUROS.....	86
7 - REFERÊNCIAS BIBLIOGRÁFICAS.....	88
APÊNDICE I.....	91

Lista de Figuras

FIGURA 1: Sistema Interativo dos Mecanismos de Causa e Efeitos (Realimentação) Envolvidos no Clima	19
FIGURA 2: Exemplo de processo estacionário e não estacionário.....	21
FIGURA 3: Transformada Breve de Fourier (utilizando função Gaussiana).....	25
FIGURA 4: Decomposição de um sinal em várias escalas.....	32
FIGURA 5: Wavelet de Haar.....	33
FIGURA 6: Wavelets de Symlet (sym2), Coiflet (coif2) e Daubechies (db2).....	34
FIGURA 7: Sinal com o complexo QRS	37
FIGURA 8: Esquema Padrão de uma Rede Neural	40
FIGURA 9: Morfologia de um Neurônio	40
FIGURA 10: Vários modelos de neurônios interconectados	42
FIGURA 11: Funcionamento de um neurônio artificial.....	43
FIGURA 12: Exemplos de Funções de Ativação	44
FIGURA 13: Aprendizagem por reforço.....	48
FIGURA 14: Representação geométrica do espaço de entrada.....	51
FIGURA 15: Separabilidade linear.....	54
FIGURA 16: Estrutura de uma rede de perceptrons de múltiplas camadas.....	55
FIGURA 17: Estrutura da Rede Neural utilizada no modelo para testes com entrada de oito elementos.....	58
FIGURA 18: Série histórica das leituras de vazão do Rio Itajaí Açu, de fevereiro de 1970 a fevereiro de 2007, efetuadas na cidade de Rio do Sul.....	59
FIGURA 19: Sobreposição de janela na série temporal.....	59
FIGURA 20: Funcionamento do modelo proposto (do janelamento de dados da série temporal a inferência por parte da Rede Neural Artificial.....	63
FIGURA 21: Diagrama de caixas com os erros quadráticos das 75 execuções agrupadas	64
FIGURA 22: Diagrama de caixas com os erros quadráticos das execuções agrupadas sem outliers.....	65
FIGURA 23: Erro médio apresentado em cada caso.....	65
FIGURA 24: Acerto percentual médio apresentado para cada wavelet.....	66
FIGURA 25: Combinação Janela / Intercalação utilizada.....	67
FIGURA 26: Erro Quadrático Apresentado pela Rede Neural Sem Wavelets.....	67
FIGURA 27: Erro Percentual Apresentado pela Rede Neural Sem Wavelets.....	68
FIGURA 28: Tempo Médio (segundos) para Execução do Experimento.....	68
FIGURA 29: Tempo Médio (segundos) para Execução das Etapas (ET1 ... ET4) com Redes Neurais Artificiais.....	69
FIGURA 30: Erro Quadrático Médio – Redes Neurais – Wavelet Chma6 – Primeira Decomposição.....	70
FIGURA 31: Acerto Percentual Médio – Redes Neurais – Wavelet Chma6 – Primeira Decomposição.....	70
FIGURA 32: Tempo Médio (segundos) para Execução do Experimento.....	71
FIGURA 33: Tempo Médio (segundos) para Execução das Etapas (ET1 ... ET4) Redes	

Neurais Artificiais com Pré-Processamento Wavelet Chma6 – Primeira Decomposição	72
FIGURA 34: Erro Quadrático Médio – Redes Neurais – Wavelet Chma6 – Segunda Decomposição.....	73
FIGURA 35: Acerto Percentual Médio – Redes Neurais – Wavelet Chma6 – Segunda Decomposição.....	73
FIGURA 36: Tempo Médio (segundos) para Execução do Experimento.....	74
FIGURA 37: Tempo Médio (segundos) para Execução das Etapas (ET1 ... ET4) Redes Neurais Artificiais com Pré-Processamento Wavelet Chma6 – Segunda Decomposição	75
FIGURA 38: Erro Quadrático Médio – Redes Neurais – Wavelet Chma6 – Terceira Decomposição.....	76
FIGURA 39: Acerto Percentual Médio – Redes Neurais – Wavelet Chma6 – Terceira Decomposição.....	76
FIGURA 40: Tempo Médio (segundos) para Execução do Experimento.....	77
FIGURA 41: Tempo Médio (segundos) para Execução das Etapas (ET1 ... ET4) Redes Neurais Artificiais com Pré-Processamento Wavelet Chma6 – Terceira Decomposição	78
FIGURA 42: Tempo médio para cada configuração do experimento utilizando conjunto de dados de 1500 , 3000 e 6000 leituras, treinados a 10, 20, 30 e 40 épocas, com janelamento de 16 leituras e intercalamento de 1.....	79
FIGURA 43: Tempo médio (segundos) para cada configuração do experimento utilizando conjunto de dados de 1500 , 3000 e 6000 leituras, treinados a 10, 20, 30 e 40 épocas, com janelamento de 16 leituras e intercalamento de 1 pré-processado por Chroma6.....	80
FIGURA 44: Tempo médio para cada configuração do experimento utilizando conjunto de dados de 1500 , 3000 e 6000 leituras, treinados a 10, 20, 30 e 40 épocas, com janelamento de 32 leituras e intercalamento de 2.....	80
FIGURA 45: Tempo médio (segundos) para cada configuração do experimento utilizando conjunto de dados de 1500 , 3000 e 6000 leituras, treinados a 10, 20, 30 e 40 épocas, com janelamento de 32 leituras e intercalamento de 2 pré-processado por Chroma6.....	80
FIGURA 46: Análise de variância efetuada sobre as 75 repetições para cada wavelet..	82

RESUMO

O presente trabalho apresenta um modelo híbrido wavelet/redes neurais, onde é empregada a análise wavelet como pré-processamento para a rede neural. Desta forma a análise desempenhará papel de extrator de características. Baseado nas características extraídas, a rede neural artificial efetua a aproximação de função com o intuito de retornar valores de previsão da série temporal desejada. Como validador para o modelo é utilizada a série temporal da vazão do Rio Itajaí Açu. Para a escolha da wavelet, a ser aplicado na quantização do modelo, são elencadas vinte candidatas (haar, db3, db5, db7, db10, db15, db20, coif1, coif2, coif3, coif4, coif5, sym3, sym5, sym7, sym10, sym15, sym20, chroma4 e chroma6). Testes repetitivos são efetuados tirando índices de erro para selecionar a melhor candidata. Baseado sob a wavelet selecionada para a resolução do problema de previsão é feito uma análise de combinações de janela/sobreposição para averiguar o comportamento do modelo utilizando diferentes parâmetros. Como resultado, obtiveram-se índices de acertos relativamente parecidos utilizando uma quantidade de dados de inferência menor, devido à característica de transformação das wavelets. Em outros casos é constatado um ganho computacional, o que é discutido separando o processo em quatro etapas distintas e simulando várias configurações para um mesmo exemplo.

Palavras-Chave: Wavelet, Redes Neurais Artificiais, Previsão

ABSTRACT

The present work presents a hybrid model wavelet/neural networks, which is employed wavelet analysis as a pre-processing work for the artificial neural network. This pre-processing activity will cast as a feature extraction. In these features extracted the neural network makes the approximation with the aim of returning values of the time series forecasting desired. To validate the model is used the Itajaí Açu river's flow history. Regarding the wavelet's choice, twenty candidates were enrolled in order to be applied in the quantization of the model, as follows: haar, db3, db5, db7, db10, db15, db20, coif1, coif2, coif3, coif4, coif5, sym3, sym5, sym7, sym10, sym15, sym20, chroma4 and chroma6. Repetitive tests are performed to extract error rates for selecting the best candidate. Based on the selected wavelet to solve the problem of forecasting time series, an analysis is made using a combination of window size/overlap to investigate the behavior using different parameters. The results obtained are very similar using a lower amount of data due to the characteristics of the wavelet transform. In other cases a computational gain is obtained which is discussed separating the process into four distinct stages and simulating various configurations for the same arrangement.

Key-Words: Wavelet, Artificial Neural Networks, Forecasting

1 INTRODUÇÃO

A inteligência é definida de várias formas, mas talvez a que mais expresse seu sentido, de acordo com pesquisadores, é sua tênua linha de divisão com a sobrevivência, onde o mecanismo é estimulado, fazendo com que aprenda e conseqüentemente se adapte para sobreviver. O desejo do homem de construir algo que exibisse um comportamento inteligente fez com que nascesse uma linha de pesquisa nomeada Inteligência Artificial.

Um dos temas estudados dentro desta grande área é o tema da inteligência artificial conexionista, que aborda de forma computacional as redes neurais (redes neurais artificiais). Estas tiveram seu desenvolvimento iniciado com a visualização do neurônio (Camillo Golgi), passando por sua modelagem artificial afim de ser utilizado computacionalmente (McCulloch & Pitts). Melhorias foram incorporadas (Frank Rosenblatt), mas limitações quanto a separação não linear criaram um período de inércia no desenvolvimento acerca do tema. Descobertas incrementais possibilitaram seu uso em problemas não linearmente separáveis e outras mais difundiram seu uso, onde sempre é tido como referência uma comparação com a psicanálise e a neurofisiologia.

Outra característica interessante tratada na inteligência artificial é a extração de características, operação muito bem desempenhada pelo ser humano. Esta atividade visa efetuar de forma útil um resumo de grande quantidade de dados para que estes sejam processados de forma satisfatória. Uma ferramenta utilizada para efetuar esta tarefa é a *wavelet*, que foi introduzida por Morlet, que efetuava análises geofísicas com o método. Formalizado anos mais tarde veio a ser conhecido como *wavelet transform*, agregando colaborações de Grossmann com a transformada inversa, Meyer com a percepção da redescoberta da fórmula de A. Calderón por Morlet e Grossmann, Mallat incrementando a análise de multiresolução à ferramenta, Daubechies trabalhando com suas wavelets assimétricas (DAUBECHIES 1996).

Considerando que a aproximação de função é uma classe de problemas que pode ser resolvido aplicando-se redes neurais artificiais (HAYKIN 2001), e a versatilidade das *wavelets* na análise de séries temporais no espectro de tempo-frequência como pode-se ver em MALLAT (1999), surge a idéia da aplicação de wavelets/redes neurais em um

modelo preditivo para séries temporais não estacionárias, onde foi tomado como validação uma série temporal de vazão fluvial.

Para *wavelets*, que é caracterizado como uma ferramenta de transformação onde a análise ocorre no escalograma de tempo-frequência (DAUBECHIES 1992), considerando que a transformação é um processo que resulta na representação dos dados de entrada de outra forma e tomando como princípio que a cada uma das representações em particular ressaltam informações específicas deste sinal (MARR 1982), toma-se esta técnica como ferramenta para quantização.

A ferramenta de Redes Neurais Artificiais dentre outras tarefas que executa com sucesso está a classificação, podendo esta ser linear ou não (HAYKIN 2001), o que a torna ferramenta ideal para o presente modelo.

O objetivo é a criação de um modelo wavelet/redes neurais, aplicando wavelets como uma atividade de pré-processamento de dados, que terá por finalidade a extração de características do sinal, onde então as redes neurais atuam como aproximadoras de função no estudo do comportamento original da série temporal.

Para executar a validação que compreende a eficiência e o aferimento de dados sobre o comportamento do modelo, neste trabalho foi utilizado a atividade de vazão fluvial do Rio Itajaí Açu, por ser a vazão resultado do comportamento climático que tem um comportamento complexo de fatores de causa e efeito (KELLOGG & SCHNEIDER 1974).

Outros objetivos incorporados ao trabalho englobam a verificação da eficácia das wavelets quando aplicadas como extratoras de características, a diferença entre a aplicação de diferentes tipos de wavelets para desempenhar a função de extratoras de características e se existe alguma diferença entre o nível de resolução trabalhado para a extração das características da série temporal e a composição do modelo final com seu desempenho, uma vez que se trabalha com as características do sinal e não com o próprio sinal.

O presente trabalho abordará métodos de modelagem computacional de séries temporais, mais especificamente comportamento de recursos fluviais, o que torna esta pesquisa interessante por apresentar um objeto de estudo cuja modelagem computacional é complexa em virtude do comportamento climático (KELLOGG &

SCHNEIDER 1974). Desta forma faz-se uma tradução do modelo comportamental natural para um modelo computacional. Como resultado tem-se ainda reflexo social neste âmbito, uma vez que melhorando ou simplificando os modelos de previsão de recursos fluviais a tomada de decisão também é melhorada.

1.1 ORGANIZAÇÃO DO TRABALHO

O trabalho está organizado em seis capítulos cujo conteúdo é brevemente descrito na lista que segue.

- Capítulo 1 - *Introdução* - procura introduzir a idéia a ser pesquisada no presente trabalho, expondo as tecnologias a serem empregadas bem como a motivação, objetivos geral e específicos, e a organização do trabalho.
 - Capítulo 2 - *Referencial Teórico* - embasamento para realizar a pesquisa relacionada aos objetivos definidos no capítulo um. Este capítulo está subdividido em quatro partes:
 - Climatologia / Meteorologia - visa oportunizar uma breve introdução sobre o problema utilizado como validador do modelo definido como objetivo;
 - Fourier - apresenta um referencial para a análise de Fourier, método que é usado para fundamentar a utilização da análise Wavelet, provendo embasamento para comparações;
 - Wavelet - dispõe o embasamento necessário para o desenvolvimento e análise dos dados apresentados nos capítulos de resultados para qualificar a extração de características, método qual a wavelet é responsável no modelo;
 - Redes Neurais Artificiais - seguindo a mesma função da fundamentação de wavelets, este visa oportunizar ao leitor o entendimento da tecnologia e sua funcionalidade para atingir a atividade de aproximação de função;
 - Capítulo 3 - *Material e Método* - demonstra como a pesquisa foi realizada, com detalhes de configuração, ferramentas e algoritmos, bem como os dados do problema para validação da pesquisa e sua fonte;
 - Capítulo 4 - *Resultados* - demonstra de forma geral os resultados obtidos na pesquisa, apresentando gráficos, resumindo os dados de execução da pesquisa

bem como o fluxo das tarefas executadas;

- Capítulo 5 - *Discussão* - efetua uma discussão em torno do problema de pesquisa junto aos resultados obtidos pelo método implementado, visando desta forma prover a análise das execuções documentadas junto ao capítulo quatro;
- Capítulo 6 - *Conclusões* - expõe de forma clara, lógica e objetiva as conclusões extraídas do presente trabalho e faz relação aos objetivos propostos na introdução.

2 REFERENCIAL TEÓRICO

2.1 CLIMATOLOGIA / METEOROLOGIA

A climatologia, de acordo com VIANELLO & ALVES (2000), significa o estudo do clima. O tema também tem origem grega, civilização que deixou uma herança rica sob tal assunto, por ter fundamentado que a diferença climática atuava na diferença de localização geográfica (do norte para o sul).

Seguindo o raciocínio acima, VIANELLO & ALVES (2000) expõe também que a partir de 1450, a atuação climática não se restringia apenas a cinturões latitudinais, mas também por áreas irregulares, sendo ela afetada também por circulação da atmosfera e pela distribuição e configuração dos oceanos e continentes.

Uma pequena retrospectiva é feita por VIANELLO & ALVES:

No século XVIII Cotes e Simpson introduziram o uso da média aritmética para estudar as marchas diárias e anual dos elementos climáticos, numa tentativa de eliminar erros nas medições físicas e astronômicas.

O enfoque cartográfico em Climatologia foi introduzido por von Humbolt, em 1817, que preparou mapas das isotermas anuais para muitas regiões do Hemisfério Norte. Introduziu-se, à época, a prática da redução das temperaturas ao nível do mar, para excluir efeitos locais. A introdução da análise dos ventos nos estudos climáticos ocorreu em 1820 por von Buch. Naquela mesma época, introduziram-se também os conceitos de massas de ar associados com o escoamento e as influências climáticas. Em 1930 Schmidt derivou equações para expressar os ganhos e as perdas de calor pela superfície da Terra exposta à radiação solar. A teoria da convecção aplicada aos ciclones abriu caminho para o entendimento do calor latente como fonte principal de energia. O primeiro mapa mundial de precipitação, usando a técnica de sombreamento, foi produzida por Berghaus em 1845. Entre 1848 e 1860 Maury produziu a primeira carta dos ventos para as regiões oceanicas, a partir de registros efetuados por navios. (2000, p. 377)

Conforme VIANELLO & ALVES (2000, p. 378): “*o tempo meteorológico é algo que varia muito sobre a face da Terra*”. Enquanto a climatologia é a configuração atmosférica aplicada a uma região específica, o que se pode definir a partir das citações de VIANELLO & ALVES (2000), a meteorologia é a caracterização instantânea da atmosfera.

De acordo com VIANELLO & ALVES: “*o tempo meteorológico é uma experiência diária: é o estado instantâneo da atmosfera, enquanto a caracterização do clima já é mais abstrata*” (2000, p. 378).

Atualmente, a Meteorologia é reconhecida como a ciência no seu sentido mais amplo. Seus interesses incluem a física, química e a dinâmica da atmosfera, bem como os efeitos dinâmicos sobre a superfície da Terra e os oceanos e sobre a vida em geral. Seus objetivos visam ao completo entendimento dos fenômenos atmosféricos, sua previsão precisa e ao controle artificial. Por isso é um dos campos mais complexos das ciências naturais e aplicadas.

A climatologia é o estudo científico do clima, interessando-se particularmente pelas aplicações práticas. Utiliza-se dos mesmos dados básicos da Meteorologia, e seus resultados são largamente usados, pelos próprios meteorologistas, nas previsões do tempo, nas aplicações na indústria, na agricultura, nos transportes, na arquitetura na biologia, na medicina etc. O intuito da climatologia é descobrir, explicar e explorar o comportamento normal dos fenômenos atmosféricos, visando ao benefício do homem, tendo em mente a irregularidade dos fenômenos são as regras gerais e não exceções. (VIANELLO & ALVES 2000, p. 379)

2.1.1 Complexidade do Clima

Os fatores climáticos estão amplamente relacionados resultando em um sistema complexo, dinâmico e realimentado (VIANELLO & ALVES 2000). Tais características podem ser apreciadas no exemplo também apresentado pelo autor:

[...] Por exemplo: o acréscimo de calor para o ambiente, em razão das atividades humanas, leva a supor um aumento de temperatura, o que poderia ser estimado pela comparação da ordem de grandeza da energia térmica acrescentada com o da radiação solar incidente. Por outro lado, o acréscimo de energia no sistema produzirá um aumento na evaporação e, conseqüentemente, na formação de nuvens. Aumentando a quantidade de nuvens, ocorrerá um bloqueio na chegada de energia solar e, em conseqüência, haverá a tendência de queda de temperatura. O calor acrescentado à atmosfera pelo homem, atingindo as regiões cobertas de neve provocará o derretimento parcial da neve, o que conduzirá a uma maior absorção de radiação solar à superfície (porque a neve possui elevada refletividade – grande albedo) e, conseqüentemente a uma elevação da temperatura. (VIANELLO & ALVES 2000, p. 381, 382)

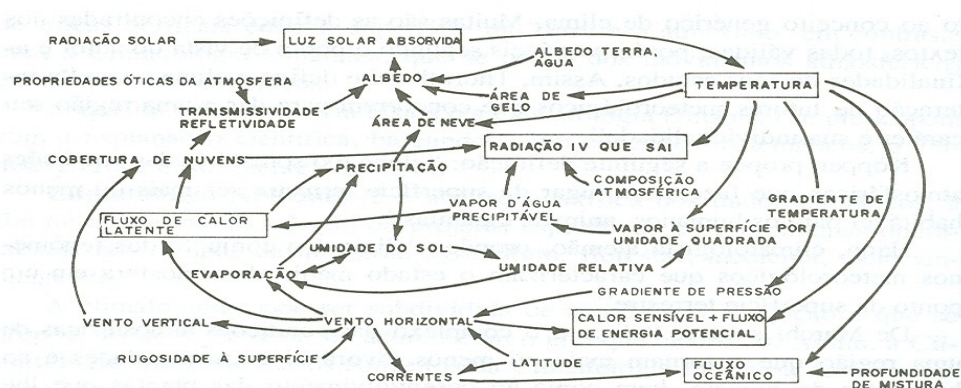


FIGURA 1: Sistema Interativo dos Mecanismos de Causa e Efeitos (Realimentação) Envolvidos no Clima

Fonte: KELLOGG & SCHNEIDER 1974

Para ilustrar a complexidade deste sistema, KELLOGG & SCHNEIDER 1974 propuseram um esquema complexo de causa e efeito apresentado na Fig. 1.

As definições citadas por VIANELLO & ALVES (2000) vão ao encontro da idéia exposta por TUCCI (2002) que define como desafio um método para conhecer adequadamente o comportamento do clima e seus reflexos.

2.1.2 Modelos de Previsão para Cheias

CANHOLI (2005) define:

Os sistemas de previsão e alerta visam evitar o fator surpresa, que muitas vezes provoca vítimas fatais e grandes prejuízos pelo alagamento de vias, aprisionamento de veículos, inundação de edificações e de equipamentos. O sistema de alerta facilita as ações preventivas de isolamento ou retirada de pessoas e de bens das áreas sujeitas a inundações, bem como a adoção de desvios de tráfego. (2005, p. 26)

FRANK & PINHEIRO (2003) dividem os modelos de previsão em dois grupos. O primeiro grupo faz referência aos modelos conceituais, onde o conhecimento físico da bacia é exigido, e o segundo grupo refere-se aos métodos empíricos, que fazem a inferência sobre dados observados criando um mapeamento entrada-saída de variáveis.

No grupo dos empíricos TUCCI, MARENGO & DIAS (2005) ainda divide-os em dois outros grupos:

Os modelos empíricos de previsão (isto é, modelos que não incluem descrições explícitas dos processos físicos envolvidos na conversão da precipitação para vazão) são basicamente de duas classes. Os modelos da primeira classe procuram obter estimativas da vazão futura, a partir do comportamento estatístico das séries de vazão no recém passado; um exemplo de modelos deste tipo são os modelos da família ARMA (*“Auto-Regression Moving-Average”*) atualmente muito usada no Brasil para a previsão de vazões. Modelos deste tipo têm a desvantagem de não aproveitar as informações sobre a quantidade e distribuição temporal da precipitação, ou outras informações climatológicas; portanto, modelos desta primeira classe não são utilizados neste trabalho.

A segunda classe de modelos empíricos procura incluir as informações sobre a precipitação e outras variáveis correlacionadas com a vazão; um exemplo desta classe de modelos são os modelos Função de Transferência (*“Transfer Function Models”*), extensivamente utilizados para previsão das saídas (*“outputs”*) de processos industriais. (2005, p. 49)

2.1.3 Objeto para Validação do Modelo Proposto

De acordo com FRANK e PINHEIRO (2003), em decorrência da grande enchente ocorrida em 1983 no vale do itajaí, surgiu a necessidade da publicação de informação de

monitoração e previsão hidrológica. Oficialmente foram implantados três sistemas de monitoramento, o primeiro ficou em vigência de 1984 a 1996 (vinculado ao governo federal). De 1996 a 1999 o período de transição, e de 2000 até o presente momento está vinculado ao governo estadual.

2.2 FOURIER

De acordo com GRAPS (1995) Joseph Fourier (1807) criou uma ferramenta importante para análise de dados, baseando-se na sua teoria de análise de frequência. Hoje a transformada de Fourier é amplamente aplicada em áreas como: sistemas lineares, teoria da probabilidade, física quântica e processamento digital de sinais, como pode-se constatar em REZA (1999).

Inicialmente a transformada fora desenvolvida para trabalhar com sinais contínuos, mas a aplicação da mesma em tecnologia digital resultou na adequação do método para trabalho com sinais discretizados, bem como o desenvolvimento de técnicas para computação do método de forma mais rápida, resultando no algoritmo da transformada rápida de Fourier (*Fast Fourier Transform*).

A utilização da transformada de Fourier consiste na aproximação de uma função a uma combinação de componentes senoidais, como trata MORETTIN (1999):

A análise de Fourier, ou análise harmônica, tem sido usada tradicionalmente para resolver algumas equações diferenciais parciais que aparecem na física matemática, como a equação do calor e a equação de ondas.

O objetivo básico é de aproximar uma função $f(t)$ por uma combinação linear de componentes senoidais, cada uma com dada frequência. O conjunto $\{w_n(t) = e^{int}, n = 0, \pm 1, \dots\}$ de funções ortogonais, de período 2π , forma a base para a análise de Fourier. Na realidade, esse conjunto é gerado por dilatações de uma única função, $w(t) = e^{it}$, ou seja, $w_n(t) = w(nt)$ para qualquer n inteiro.

O fato básico é que toda função periódica, de período 2π , de quadrado integrável, é gerada por uma superposição de dilatações inteiras da função $w(t)$. (2000, p. 31)

A utilidade da transformada de Fourier está na sua habilidade de analisar um sinal no domínio do tempo pelo domínio da frequência como afirma GRAPS (1995). A transformada de Fourier inicialmente tinha suas limitações, e uma delas era a aplicação da mesma a sinais não estacionários. Uma característica presente em um sinal não estacionário onde sua frequência básica é variável de acordo com o tempo, como pode ser observada na Fig. 2.

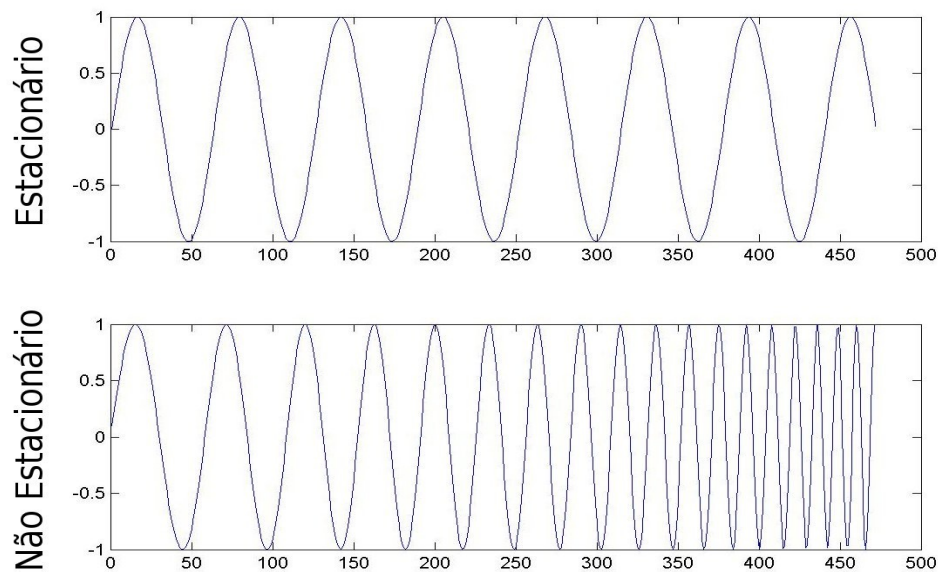


FIGURA 2: Exemplo de processo estacionário e não estacionário.

Com a formulação da transformada breve de Fourier, que pode ser visto nos trabalhos de WELCH (1967) e no trabalho de GRIFFIN (1983), torna-se possível a implementação da transformada a sinais não estacionários como o processamento do sinal de fala.

2.2.1 Transformada de Fourier

Essencialmente a transformada de Fourier faz a extração sucessiva das senoides que compõe um sinal temporal, extraíndo suas frequências e amplitudes, dando a característica de representação da função em domínio de frequência como comenta REZA (1999).

Para extração do espectro de frequência é utilizada então a equação da transformada que é dada por:

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt \quad (1)$$

Onde j vale $\sqrt{-1}$, $x(t)$ é a função contínua no tempo e $X(f)$ é a sua correspondência na transformada. Conforme REZA (1999), a transformada deve ser aplicada para sinais onde a energia é finita, característica satisfeita pela inequação:

$$\int_{-\infty}^{\infty} |x(t)|^2 dt < \infty \quad (2)$$

Mas em casos contínuos a energia não é finita, o que leva a considerar somente o período do sinal. Supondo que o período seja T (verificar se existe uma razão entre período e frequência) e de frequência $f_0=1/T$, assim $x(t)$ satisfaz $x(t)=x(t+T) \forall t$, assim se a energia for finita, pode então ser expressado por uma seqüência de combinações harmônicas, obtidas a partir da extração dos coeficientes utilizando a equação:

$$c_k = \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t) e^{-j2\pi k f_0 t} dt \quad (3)$$

E a condição de admissibilidade, que comprova a energia finita do período fica:

$$\frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} |x(t)|^2 dt < \infty \quad (4)$$

2.2.2 Transformada Inversa

Para os mesmos moldes da transformada existe a inversa afim de montar o sinal caracterizado pelo espectro de frequência gerado. Desta forma tem-se que a inversa da transformada é expressada por:

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi f t} df \quad (5)$$

E a equação inversa para séries de Fourier onde a periodicidade é considerada (para sinais contínuos):

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{j2\pi k f_0 t} \quad (6)$$

2.2.3 Transformada Discreta de Fourier

Avanços na tecnologia digital, como explana REZA (1999), resultaram na necessidade de trabalho com sinais discretos, o que serviu como ponto para modificações na transformada de Fourier, possibilitando então, a partir daquele momento, a transformada em sinais discretos. A transformada é apresentada pela

equação:

$$X(e^{j2\pi f}) = \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi n f} \quad (7)$$

E sua inversa:

$$x[n] = \int_{-\pi}^{\pi} X(e^{j2\pi f}) e^{j2\pi n f} df \quad (8)$$

Onde $x[n]$ é a função discreta. A transformada é contínua e periódica no domínio da frequência, apresentando período 2π .

Como a computação destas séries deve ser feita utilizando seqüências finitas e discretas, as necessidades resultaram em outra formulação que é apresentada por:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi n \frac{k}{N}} \quad \text{for } k=0,1,2,\dots,N-1 \quad (9)$$

E sua inversa:

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j2\pi n \frac{k}{N}} \quad \text{for } k=0,1,2,\dots,N-1 \quad (10)$$

Onde $x[n]$ é apresentado como uma seqüência periódica de período N . Onde $x[n] = x[n+N] \forall n$.

2.2.4 Transformada Rápida de Fourier (FFT – Fast Fourier Transform)

A complexidade computacional existente no cálculo da transformada discreta de Fourier levou a criação da transformada rápida de Fourier que pode facilitar tanto a transformação quanto sua inversa, assim como comenta MERTINS:

For a complex-valued input signal $x(n)$ of length N the implementation of the DFT matrix W requires N^2 complex multiplications. The idea behind the fast Fourier transform (FFT) is to factorize W into a product of sparse matrices that altogether require a lower implementation cost than the direct DFT. Thus, the FFT is a fast implementation of the DFT rather than a different transform with different properties. (1999, p. 85)

MALLAT comenta sobre a construção do algoritmo de transformação rápida de Fourier e da complexidade computacional resultada do algoritmo:

For a signal f of N points, a direct calculation of the N discrete fourier sums

$$f[k] = \sum_{n=0}^{N-1} f[n] \exp\left(\frac{-i2\pi kn}{N}\right), \text{ for } 0 \leq k < N \quad (3.37)$$

requires N^2 complex multiplication and additions. The fast Fourier transform (FFT) algorithm reduce the numerical complexity to $O(N \log_2 N)$ by reorganizing the calculations.

When the frequency index is even, we group the terms n and $n + N/2$

$$f[2k] = \sum_{n=0}^{N/2-1} (f[n] + f[n + \frac{N}{2}]) \exp\left(\frac{-i\pi kn}{N}\right) \quad (3.38)$$

When the frequency index is odd, the same grouping becomes

$$f[2k+1] = \sum_{n=0}^{N/2-1} \exp\left(\frac{-i2\pi n}{N}\right) (f[n] - f[n + \frac{N}{2}]) \exp\left(\frac{-i\pi kn}{N/2}\right) \quad (3.39)$$

Equation 3.38 proves that the even frequencies are obtained by calculating the discrete fourier transform of $N/2$ periodic signal.

$$f_e[n] = f[n] + f[n + \frac{N}{2}]$$

Odd frequencies are derived from (3.39) by computing the fourier transform of the $N/2$ periodic signal

$$f_o[k] = \exp\left(\frac{-i2\pi n}{N}\right) (f[n] - f[n + \frac{N}{2}])$$

A discrete Fourier transform of size N may thus be calculated with two discrete Fourier transform of size $N/2$ $O(n)$ operations. (1999, p. 57)

Para a inversa da transformada rápida de Fourier, de acordo com MALLAT (1999) tem-se:

$$f^*[n] = \frac{1}{N} \sum_{k=0}^{N-1} f^*[k] \exp\left(\frac{-i\pi kn}{N}\right) \quad (11)$$

Pode-se constatar ainda que existem variações do algoritmo de transformada rápida de Fourier. Conforme MALLAT(1999) o mais eficiente algoritmo para FFT é o *split-radix*, que inicialmente é mais complicado mas necessita somente $N \log_2 N$ multiplicações reais, e $3N \log_2 N$ adições.

2.2.5 Short Time Fourier Transform

A transformada breve de Fourier fora desenvolvida em 1946 por Gabor, nada mais é do que uma janela para análise da série, onde os valores dentro da janela, que pode ser

transladada, são tomados como valores para a transformada de Fourier como expõe MERTINS:

The short-time Fourier transform (STFT) is the classical method of time-frequency analysis. The concept is very simple. We multiply $x(t)$, which is to be analysed, with an analysis window and then compute the Fourier transform of the window signal:

The analysis window $\gamma^*(t-\tau)$ suppresses $x(t)$ outside a certain region, and the Fourier transform yields a local spectrum. [...] (1999, p. 196)

Para a janela pode-se utilizar diferentes funções, como Gabor que fundamentou a utilização da transformada de Fourier com Janelas utilizando a função Gaussiana. Um exemplo da transformação breve de Fourier fazendo uso da função Gaussiana como janela pode ser observado na Fig. 3.

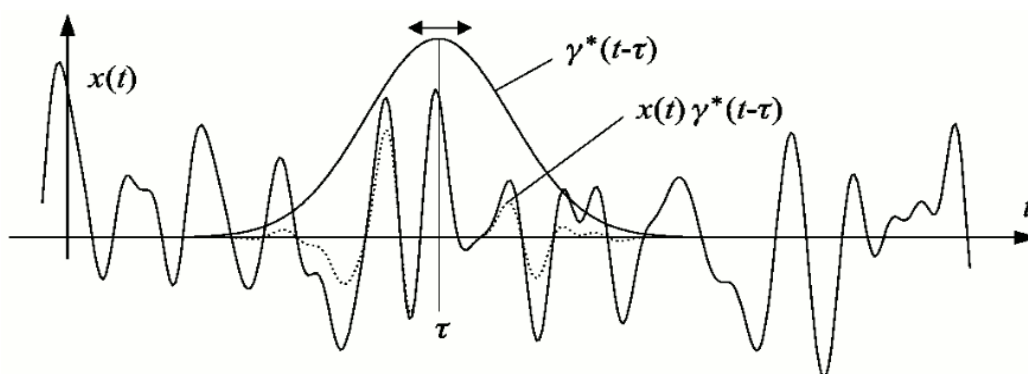


FIGURA 3: Transformada Breve de Fourier (utilizando função Gaussiana)

Fonte: MERTINS 1999, p. 197

A equação para transformada breve de Fourier é dada por:

$$F_x^\tau(\tau, \omega) = \int_{-\infty}^{\infty} x(t) \gamma^*(t-\tau) e^{-j\omega t} dt \quad (12)$$

Com a aplicação da transformada breve de Fourier é possível a aplicação da técnica introduzida por Fourier em séries de dados onde a função $f(x)$ não é totalmente conhecida, o que era inviável com a transformada de Fourier, como comentam GOSWAMI & CHAN:

Unlike the case of Fourier transform, in which the function $f(t)$ must be known for the entire time axis before its spectral component at any single frequency can be computed, STFT needs to know $f(t)$ only in the interval in which $\phi(t-b)$ is nonzero. (1999, p. 60)

Onde $\phi(t-b)$ refere-se a função que é utilizada como janela de análise.

Desta forma pode-se efetuar então análises não somente do espectro de frequência

gerado pela transformada, mas também quanto ao tempo, possibilitando a análise tempo-frequência. Esta característica possibilitou a aplicação da técnica também em sinais não estacionários.

Considerando agora a reconstrução do sinal $x(t)$ a partir de $F_x^\tau(\tau, \omega)$, MERTINS (1999) o faz aplicando a equação:

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F_x^\tau(\tau, \omega) g(t-\tau) e^{j\omega t} d\tau d\omega \quad (13)$$

2.2.6 DSTFT – Discrete Short Time Fourier Transform

De acordo com MALLAT (1999), a idéia por trás da transformada breve discreta de Fourier, segue o mesmo princípio da idéia introduzida para a transformada discreta de Fourier, mas considerada a função da janela.

A DSTFT, de acordo com MERTINS (1999), é a troca da integral definida pela transformada de Fourier por uma somatória. Que é dada por:

$$F_x^\tau(m, e^{j\omega}) = \sum_n x(n) \tau^*(n-mN) e^{-j\omega n} \quad (14)$$

E a reconstrução do sinal se dá pela equação:

$$x(n) = \sum_{-\infty}^{\infty} \sum_{k=0}^{M-1} X(m, k) g(n-mN) W_M^{-kn} \quad (15)$$

Onde τ representa a função da janela e g a função da janela de forma discreta.

2.3 WAVELETS

De acordo com GOSWAMI & CHAN (1999), a transformada de Fourier para sinais estacionários é uma ferramenta poderosa, mas levando-se em conta que para efetuar o cálculo de todas as frequências precisa-se ter em mãos o sinal completamente conhecido, e que uma perturbação em um pequeno trecho de sinal provoca uma grande mudança em escopo global, torna Fourier uma ferramenta não tão interessante para trabalho com séries não estacionárias, mesmo quando utilizado a *Short Time Fourier Transform*, pois a aplicação da STFT em um sinal onde a frequência é alterada linearmente com o tempo torna-se uma ferramenta ineficaz, uma vez que a seleção do

tamanho da janela propiciará a uma frequência específica.

A similaridade existente entre a STFT e a transformada Wavelet não é por mero acaso, de acordo com DAUBECHIES (1996), o desenvolvimento das wavelets iniciou-se com Morlet que necessitava de uma ferramenta que permitisse a análise de frequências altas e baixas ao mesmo tempo, desta forma criou um mecanismo onde a função de janela pudesse ser comprimida ou expandida. Grossmann desenvolveu a fórmula inversa à integral da transformada de Morlet, incrementando o poder da ferramenta. Meyer aponta o desenvolvimento de Morlet e Grossmann como uma redescoberta da fórmula de Calderón, que a introduziu na disciplina de Análise Harmônica, surgida da análise de Fourier. E logo após, Mallat incrementou ainda mais a ferramenta com a análise de multiresolução.

De acordo com NAKKEN tem-se: “[...] *Fourier transform breaks the signal into a series of sine waves of different frequencies, the wavelet transform breaks the signal into its ‘wavelets’: scaled and shifted versions of the ‘mother wavelet [...]’* (1999, p. 288). A utilização de wavelets também não favorece uma frequência específica por apresentar uma janela com tamanho finito sendo que esta pode variar.

De acordo com GRAPS: “*The most interesting dissimilarity between these two kinds of transforms is that individual wavelet functions are localized in space. Fourier sine and cosine are not.*” (1995, p. 6). Para a análise de sinais não estacionários seria interessante então a aplicação de um método que desse bons resultados em escala de tempo-frequência. Como citam GOSWAMI & CHAN:

Our objective is to devise a method that can give good time-frequency resolution at an arbitrary location in t - ω plane. In other words, we must have a window function whose radius increase in time (reduce in frequency) while resolving the low-frequency contents, and decreases in time (increases in frequency) while resolving the high-frequency contents of a signal. This objective leads us to the development of wavelet functions $\psi(t)$. GOSWAMI & CHAN (1999, p. 68)

Desta forma é dado a Wavelet, a característica central de uma ferramenta para manipulação de auto-similariedade como comenta WORNELL: “[...] *The Wavelet transform constitutes as natural a tool for the manipulation of self-similar or scale-invariant signal as the Fourier transform does for translation-invariant signals such as stationary, cyclostationary and periodic signals. [...]’* (1995, p. 8).

Com idéia semelhante, MORETTIN (2000) cita que as ondaletas são localizadas no tempo (ou espaço), contrariamente do que ocorre com as funções trigonométricas. Esse comportamento torna-as ideais para analisar sinais não estacionários, contendo transitoriedades e estruturas tipo fractais.

A teoria de Wavelets é relativamente nova mas as idéias que a acercam não, como fundamenta WORNELL:

[...] Ideed, wavelet theory can be reviewed as a convenient and useful mathematical framework for formalizing and relating some well-established methodologies from a diverse number of areas withing mathematics, physic and engeneering. Example includes:

- piramidal image decomposition in computer vision;
- multigrid methods in the solution of partial-differential and integral equations;
- spectrogram methods in speech recognition;
- progressive transmission algorithms and embedded coding in communications;
- multirate filtering algorithms in digital audio, speech and image coding, voice scrambling, frequency division data multiplexing and time-frequency division data cross-multiplexing. (1995, p. 9)

2.3.1 Transformada Wavelet

De acordo com WORNELL (1995) e GOSWAMI & CHAN (1999), a transformada de um sinal $x(t) \Leftrightarrow X_v^\mu$ é definida como uma projeção de $x(t)$ em uma classe de funções que são herdadas de um protótipo de função Wavelet ($\psi(t)$) (também nomeada de wavelet mãe ou *mother wavelet*) normalizadas, dilatadas e transladadas de formas diferentes entre si. Assim temos:

$$W\{x(t)\} = X_v^\mu = \int_{-\infty}^{\infty} x(t) \psi_v^\mu(t) dt$$

onde

(16)

$$\psi_v^\mu(t) = |\mu|^{-1/2} \psi\left(\frac{t-v}{\mu}\right)$$

Na equação, μ e v representam os parâmetros de dilatação e translação, e englobam valores satisfeitos por $-\infty < \mu, v < \infty, \mu \neq 0$. A condição de admissão para transformação inversa é dada por:

$$\int_{-\infty}^{\infty} |\psi(\omega)|^2 |\omega|^{-1} d\omega = C_\psi < \infty$$
(17)

onde $\psi(t)$ é a transformada Fourier da wavelet, definida como:

$$\psi(\omega) = \int_{-\infty}^{\infty} \psi(t) e^{-j\omega t} dt \quad (18)$$

que é equivalente a condição de admissão:

$$\int_{-\infty}^{\infty} \psi(t) dt = 0 \quad (19)$$

2.3.2 Transformada Inversa

Basicamente, a idéia por trás da transformada wavelet inversa é a reconstrução do sinal a partir de um sinal transformado, como descrevem TANG, LIU & YANG: “*As the same as Fourier Transform, wavelet transform is invertible. The inverse wavelet transform can be viewed as the reconstruction of the original signal. [...]*” (2000, p. 61), que pode ser efetuado sobre uma integração bidimensional sobre os parâmetros de dilatação e translação μ e ν respectivamente.

$$x(t) = W^{-1}\{X_{\nu}^{\mu}\} = \frac{1}{C_{\psi}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} X_{\nu}^{\mu} \psi_{\nu}^{\mu}(t) \mu^{-2} d\mu d\nu \quad (20)$$

Antes da formulação da equação de inversão da transformada TANG, LIU & YANG (2000) provam que a transformada wavelet mantém a lei de conservação de energia e apontam que para uma função poder ser utilizada como wavelet deve satisfazer a condição de admissibilidade exposta abaixo:

$$C_{\psi} = \frac{\int_{-\infty}^{\infty} |\hat{\psi}(\omega)|^2}{|\omega|} d\omega < \infty$$

2.3.3 Análise Multiresolução (MRA – Multiresolution Analysis)

A análise de multiresolução foi introduzida por Mallat, baseando-se sobre o trabalho de Morlet, Grossmann e Meyer (DAUBECHIES 1996). De acordo com TANG, LIU & YANG (2000) a multiresolução tem se tornado uma ferramenta importante para a área de processamento digital de sinais, processamento de imagens e reconhecimento de padrões.

MALLAT define a propriedade de multiresolução como “*Adapting the signal*

resolution allows one to process only the relevant details for a particular task” (1999, p. 221). MORETTIN (2000) define-a com um conceito ainda mais simples, “A análise de multiresolução (AMR) permite analisar os dados disponíveis em várias escalas de resolução. É como se pudéssemos ‘olhar’ esses dados por um microscópio e ver seu comportamento em várias ‘magnificações’”.

Ainda de acordo com MALLAT:

The approximation of a function f at a resolution 2^j is specified by a discrete grid of samples that provides local averages of f over neighborhoods of size proportional to 2^j . A multiresolution approximation is thus composed of embedded grids of approximation. More formally, the approximation of a function at a resolution 2^{-j} is defined as an orthogonal projection on a space $V_j \subset L^2(\mathbb{R})$. The space V_j regroups all possible approximations at resolution 2^j . The orthogonal projection of f is the function $f_j \in V_j$ that minimizes $\|f - f_j\|$. The following definition introduced by Mallat [254] and Meyer [47] specifies the mathematical properties of multiresolution spaces. To avoid confusion let us emphasize that a scale parameter 2^j is the inverse of resolution 2^j .

Definition 7.1 (Multiresolutions) A sequence $\{V_j\}_{j \in \mathbb{Z}}$ of closed subspaces of $L^2(\mathbb{R})$ is a multiresolution approximation if the following 6 properties are satisfied:

$$\forall (j, k) \in \mathbb{Z}^2, f(t) \in V_j \Leftrightarrow f(t - 2^j k) \in V_j,$$

$$\forall j \in \mathbb{Z}, V_{j+1} \subset V_j,$$

$$\forall j \in \mathbb{Z}, f(t) \in V_j \Leftrightarrow f\left(\frac{t}{2}\right) \in V_{j+1} \quad (1999, p. 221)$$

$$\lim_{j \rightarrow \infty} V_j = \bigcap_{j=-\infty}^{+\infty} V_j = \{0\},$$

$$\lim_{j \rightarrow -\infty} V_j = \text{Closure}\left(\bigcup_{j=-\infty}^{+\infty} V_j\right) = L^2(\mathbb{R})$$

Para cada propriedade citada, MALLAT (1999) define sua finalidade. A primeira refere-se quanto a variação sob qualquer translação proporcional a escala 2^j . A segunda prova que a aproximação, a resolução 2^j , se tem todos os dados necessários para computar a aproximação a uma resolução grosseira. Dilatando em 2, V_j dobra os detalhes, assim a terceira garante que é definido uma aproximação a uma resolução grosseira 2^{j-1} . A medida que a resolução 2^j tende a 0, como tratada na quarta propriedade, são perdidos os detalhes de f e, do contrário, quando a resolução 2^j tende ao infinito o sinal converge ao sinal original adicionando cada vez mais detalhes à série,

como exposto na quinta propriedade.

De acordo com GOSWAMI & CHAN (1999), para atingir a análise de multiresolução de uma função precisa-se ter uma outra função de energia-finita $f(t) \in L^2 \mathbb{R}$ denominada função de escala.

$$\psi_n^m(t) = 2^{\frac{m}{r}} \psi(2^m t - n) \quad (21)$$

Para cálculo dos coeficientes (c) e dos detalhes (d) utiliza-se as equações:

$$c_{j0,k} = \int_{-\infty}^{\infty} f(t) \phi_{j0,k}(t) dt$$

$$d_{jj,k} = \int_{-\infty}^{\infty} f(t) \psi_{j,k}(t) dt \quad (22)$$

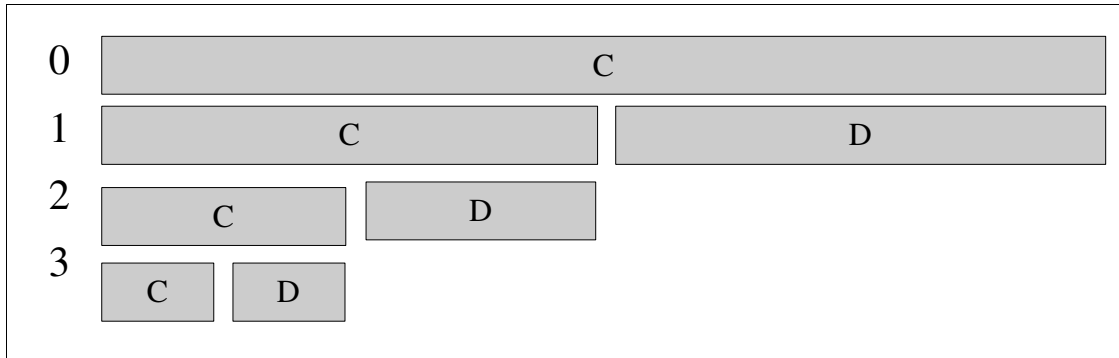


FIGURA 4: Decomposição de um sinal em várias escalas

A aplicação consecutiva das equações (22) gera como retorno os valores apresentados na Fig. 4.

A decomposição do sinal em várias escalas pode ser melhor compreendida com uma citação de SOLTANI (2002):

The multiscale decomposition uses a low- and a band-pass filters [5]. Applying this pair of filters to a time series leads to a first series which contains the trend (or slower dynamics) and a second one which contains the details (or fastest dynamics). The original time series may be reconstructed by summing up the trend and the detail series. (2002, p. 268)

De acordo com WORNELL (1995) pode-se extrair a série original a partir de qualquer escala, efetuando a soma recursiva dos conjuntos ortogonais.

2.3.4 Transformada Discreta Wavelet (DWT – Discrete Wavelet Transform)

A transformada discreta Wavelet, refere-se a uma forma de trabalho com dados discretizados sob a transformada Wavelet. WORNELL define a DWT como:

The discrete wavelet transform (DWT) refers to a discrete-time framework for implementing the orthogonal wavelet transform. The basic notion is that rather than implementing the analysis directly as a sequence of continuous time filter-and-sample operation [...], the analysis can be reformulated into a single continuous-to-discrete conversion procedure followed by some interactive discrete-time processing. Likewise, the synthesis can be reformulated from a series of conventional modulations [...] into an interactive discrete-time procedure followed by a single discrete-to-continuous conversion. (1995, p. 21)

De acordo com GRAPS (1995), a equação 23 é empregada, sendo ela dilatada, modificando seu comprimento, e transladada, modificando sua posição, utilizando os parâmetros inteiros s e l respectivamente, para criar novas versões da wavelet mãe.

$$\phi_{(s,l)}(x) = 2^{-\frac{s}{2}} \phi(2^{-s}x - l) \quad (23)$$

Desta forma aplicando a uma equação de escala (24), onde $W(x)$ é a função de escala para a wavelet mãe ϕ , e c_k são os coeficientes wavelet.

$$W(x) = \sum_{k=-1}^{N-2} (-1)^k c_{k+1} \phi(2x + k) \quad (24)$$

2.3.5 Famílias de Wavelet

Como já fundamentado uma função wavelet pode ser dilatada e transladada construindo uma família de outras versões de uma mesma função, podendo desta forma ser chamada de *mother wavelet* (wavelet mãe) por dar uma forma base a estas outras versões.

Existem várias definições de famílias wavelets na literatura. Dentre elas estão as wavelets de Haar, Morlet, Chapéu Mexicano, Daubelets, Coiflets e Symlet.

De acordo com MORETTIN (2000) o exemplo mais antigo e simples de wavelet é a função de Haar, desenvolvida em HAAR (1910). O comportamento da função se dá por:

$$\psi^{(H)}(t) = \begin{cases} 1, 0 \leq t < 1/2 \\ -1, 1/2 \leq t < 1 \\ 0, \text{otherwise} \end{cases} \quad (25)$$

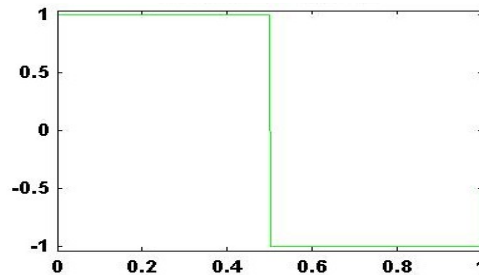


FIGURA 5: Wavelet de Haar

O resultado da aplicação da wavelet de Haar sobre uma série de dados, resulta em uma segunda série de dados composta de dois conjuntos, coeficientes e detalhes respectivamente, a partir dos quais é possível remontar a série inicial.

Daublet, Coiflet e Symlet foram desenvolvidas por Ingrid Daubechies (DAUBECHIES 1992) e são comumente utilizadas para resolução de problemas que envolvem aplicação de wavelets, como pode ser visto nos trabalhos SINGH, VASQUEZ & SINGH (1997) e SOUZA, PAGAMISSE & MENEGUETTE Jr (2005).

A representação das funções wavelet Symlet, Daublet e Coiflet estão expostas na Fig. 6.

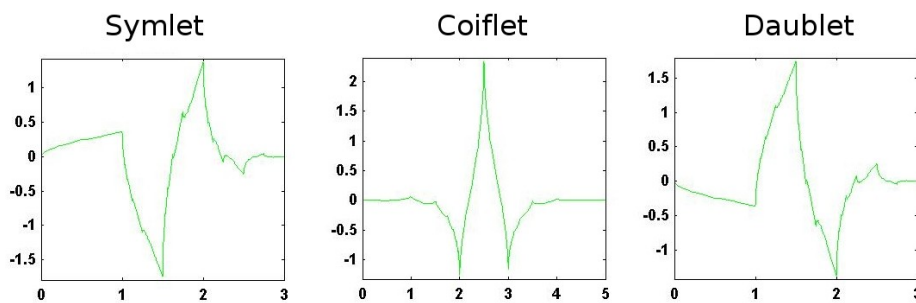


FIGURA 6: Wavelets de Symlet (sym2), Coiflet (coif2) e Daubechies (db2)

2.3.6 Aplicação de Wavelet

A utilização do modelo Wavelet se dá em vários campos, um em específico é na extração de características, pode-se notar pela fundamentação já vista que a transformada em si cria uma representação alternativa do sinal. MARR (1982) define

que uma representação específica torna certas informações explícitas em detrimento de outras e ressalta ainda que as várias formas de representação podem facilitar trabalhos diferentes com um mesmo sinal. A seguir demonstram-se algumas de suas aplicações.

2.3.6.1 Wavelets em Sinais Geofísicos

De acordo com BOTELLA, ROSA-HERRANZ & GINER (2003), os sinais geofísicos são sinais não estacionários, necessitando então ferramentas que possibilitem o trabalho com sinais desta classe, o que torna Wavelets uma opção para análise de sinais geofísicos.

Como cita KUMAR, wavelet pode ser usado em sinais geofísicos de duas formas:

Wavelets are essentially used in two ways when studying geophysical processes or signals: (1) as an integration kernel for analysis to extract information about the process and (2) as a basis for representation or characterization of the process. Evidently, in any analysis or representation the choice of the kernel or basis function determines the nature of information that can be extracted or represented about the process. This leads us to the following questions: (1) What kind of information can we extract using wavelets, and (2) how can we obtain a representation or description of a process using wavelets? (1997, p. 386)

As perguntas criadas pelo autor na citação acima são respondidas por ele no mesmo trabalho, a primeira consiste basicamente da propriedade de análise em tempo-frequência da wavelet, característica importante para sinais não estacionários, como já citado por outros autores referenciados no presente trabalho, ou então para sinais com componentes transitórios, que apresentam características em diferentes escalas ou ainda aqueles que apresentam singularidades. Wavelets então permitem um nível de detalhe maior em menores escalas e características gerais para escalas maiores.

A segunda pergunta tem sua resposta embasada na propriedade de representação de um sinal por uma sequência infinita de versões dilatadas (ou retraídas) e transladadas de uma única função, conhecida como Wavelet Mãe. Cada versão criada é também multiplicada por um coeficiente apropriado.

A representação de sinais utilizando wavelet visa a seleção de uma representação particular do sinal assim como qualquer outra transformada, e KUMAR (1997) adiciona ainda que cada problema deve ser analisado de forma independente e selecionada a wavelet cujo problema é melhor resolvido.

Detecção de Terremotos

BOTELLA, ROSA-HERRANZ e GINER (2003) discutem em seu trabalho a implementação de um novo meio para detecção de terremotos baseado numa filtragem do sinal utilizando o algoritmo da Transformada Wavelet Discreta. No modelo proposto, a filtragem seria um pré-processamento para o método que já era utilizado.

O resultado da aplicação de Wavelets (especificamente DB8 para este problema) como um pré-processamento resultou em uma detecção mais precisa de terremotos (aumento de 63,66% da precisão em relação ao modelo anterior), assim como na redução de falsos alarmes (redução de 6,91% em relação ao modelo anterior).

Análise Climática

LIN, BIAN & YOU (1996), em seu trabalho desenvolveram a análise da temperatura dos últimos 500 anos na China. A partir de seu estudo conseguiu constatar a versatilidade do método Wavelet. Segundo LIN: *“Wavelet analysis is a powerful mathematical method applied widely in many fields. Because wavelet analysis has better local properties than other methods such as Fourier transformation, it may clearly identify singularities in a signal”* (1996, p. 25).

Em seu trabalho também é concluído que wavelets podem ser utilizadas para extração de curvas características, as quais utilizadas como parâmetros junto a modelos matemáticos possibilitam a projeção de valores futuros, criando assim um mecanismo de previsão. Este mecanismo pode informar valores futuros ou ainda novos pontos de catástrofes.

2.3.6.2 Wavelets em Biomedicina

UNSER & ALDROUBI (1996) fundamentam que a aplicação de novos modelos, diferentes das tradicionais técnicas de Fourier para análise, onde a informação é localizada no plano tempo-frequência, o que permite a troca de resolução é interessante para sinais não estacionários. E na Engenharia Biomédica este tipo de análise é necessária.

De acordo com UNSER & ALDROUBI (1996), modelos como estes podem ser

aplicados em vários problemas comuns, dentre eles cita: eletrocardiograma e ressonância magnética. Mas cada caso precisa ser tratado individualmente pelas suas particularidades. Na medicina, diferente de engenharia, sinais podem conter a informação relevante em escalas diferentes, desta forma é necessário modelos robustos que trabalham na maioria das circunstâncias, enquanto na engenharia o problema é bem definido e sabe-se donde deve-se extrair a informação.

Outro aspecto importante para a biomedicina, também tratado por UNSER & ALDROUBI (1996), é de que a informação de interesse pode ser uma combinação de características localizadas temporalmente ou espacialmente como por exemplo: picos, fases transitórias ou ainda microcalcificações em mamografias, estes problemas necessitam de uma técnica suficientemente versátil para manipular eventos que podem ser extremos em termos da sua localização tempo-frequência.

Bio-Acústica

A extração de sinais sonoros cardíacos podem ser efetuados através de um estetoscópio ou microfones posicionados dentro do coração através de um cateter ou ainda no esôfago com a ajuda de um tubo.

UNSER & ALDROUBI (1996) expõem uma linha de desenvolvimento da utilização da transformada na bio-acústica. Baseado nisto KHADRA & DICKHAUS (1993), definiram Wavelet como um método eficaz para análise e caracterização dos sons primários do coração. Seguindo o mesmo problema, Obaidat que fez uso da transformada para extração de componentes do segundo som do coração, o que não é possível ser feito com a utilização de outro método.

Eletrocardiografia

Segundo UNSER & ALDROUBI (1996), a eletrocardiografia representa a mudança de potencial elétrico durante o ciclo cardíaco que é extraído por eletrodos espalhados no corpo. A Fig. 7 demonstra um exemplo de sinal obtido para uma eletrocardiografia.

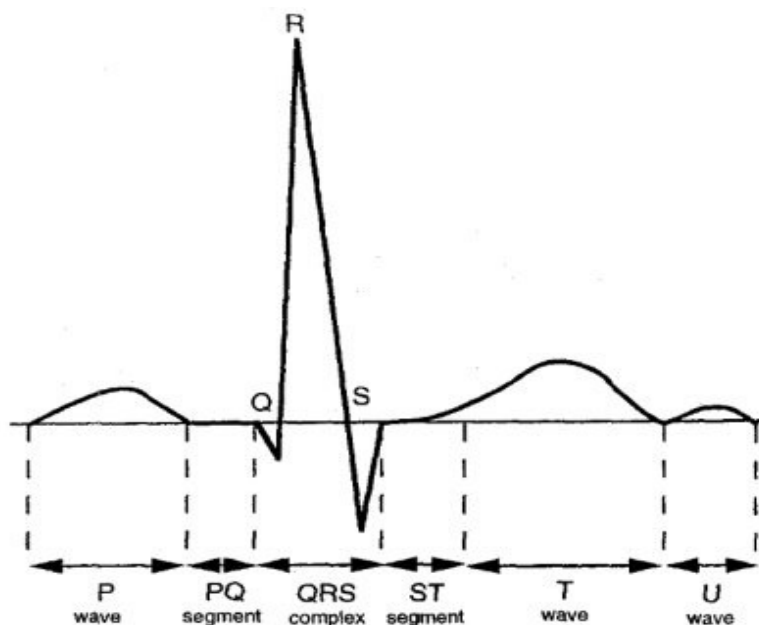


FIGURA 7: Sinal com o complexo QRS

Fonte: UNSER & ALDROUBI (1996, p. 631)

Pode-se ver que existe a divisão em áreas, onde o problema reside na localização do complexo QRS. Wavelets neste caso, aplicado por SENHADJI & CARRAULT (1995), mostrou um excelente resultado.

Eletroencefalografia

Outra área também beneficiada pela utilização de Wavelets é a eletroencefalografia, que tem como uma das principais aplicações a detecção de epilepsia. Sinais transitórios que se desenvolvem lentamente a medida do tempo em uma amplitude elevada, refletindo descarga anormal de um grande grupo de neurônios. O formato e tamanho destas ondas pode variar substancialmente de paciente para paciente.

A Wavelet neste caso se tornou uma ferramenta de detecção válida pela natureza mista do fenômeno, atuando então como extrator de características no trabalho desenvolvido por KALAYCI & OZDAMAR (1995).

2.4 REDES NEURAIAS ARTIFICIAIS (ANN – Artificial Neural Networks)

HAYKIN (2001), assim como FREEMAN & SKAPURA (1991), expõem que o desenvolvimento de redes neurais artificiais foram motivadas pelo funcionamento do cérebro, que possui uma estrutura complexa, não linear e paralela.

Para ilustração, HAYKIN (2001), toma como exemplo o reconhecimento de faces, a partir de um estímulo visual, assim que o estímulo visual é pré-processado o cérebro efetua em aproximadamente 100 a 200 ms a tarefa de classificação da face, que pode, por exemplo, reconhecer uma face familiar em um cenário totalmente não familiar.

Do ponto de vista físico, HAYKIN (2001) explica que ao nascer o ser humano pode modelar suas próprias regras baseado nas experiências por ele vivenciadas, o que podem ser ajustadas durante todo o processo de vida. Esta característica é definida como plasticidade, sinônimo de um neurônio em desenvolvimento.

Uma rede neural artificial é composta de neurônios artificiais, que são criados e interligados a fim de efetuar uma tarefa em particular executada pelo cérebro. Desta forma existem duas características comuns ao funcionamento do cérebro de acordo com HAYKIN:

Uma rede neural é um processador maciçamente paralelamente distribuído constituído de unidades de processamento simples, que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para uso. Ela se assemelha ao cérebro em dois aspectos:

1. O conhecimento é adquirido pela rede a partir de seu ambiente através de um processo de aprendizagem;
2. Forças de conexão entre neurônios, conhecidas como pesos sinápticos, são utilizadas para armazenar o conhecimento adquirido.

(2001, p. 28)

Ainda de acordo com HAYKIN (2001), o processo pelo qual o conhecimento é processado é conhecido como algoritmo de aprendizagem, este por sua vez faz as mudanças de pesos das ligações sinápticas de um neurônio com o outro na estrutura da rede neural artificial. Desta forma também é possível que a própria rede altere sua topologia, o que é motivado pelo fato que neurônios reais também morrem e novas conexões sinápticas crescem.

Alguns benefícios são alcançados com a utilização de redes neurais, são apontados por HAYKIN (2001): **não linearidade**, **mapeamento de entrada/saída** o que permite efetuar a tarefa de aprendizado baseado sobre exemplos informados; **adaptividade** capacidade de alterar seus pesos a adaptar-se ao ambiente inscrito (propriedade interessante para processos não estacionários); **resposta probatória** o que pode ser utilizado para selecionar um determinado padrão tanto como verificar a confiança depositada no padrão selecionado; **informação contextual** todo neurônio é

influenciado pela atividade global dos outros neurônios na rede, o que permite uma resposta contextual fornecida naturalmente pela rede; **tolerância a falhas** onde o dano causado a um neurônio não compromete a estrutura da rede mas uma pequena fatia da qualidade do resultado retornado pelo uso desta rede; **implementação VLSI**, a estrutura das redes neurais de forma paralela permite a sua implementação usando VLSI (very-large-scale-integrated); **análise de modelagem uniforme**, independente da estrutura de rede utilizada os termos e componentes são os mesmos facilitando o entendimento e a criação de novas teorias e por fim a **analogia à neurobiologia** o que permite a criação de novas idéias baseado sobre a observação de fenômenos na neurobiologia.

2.4.1 Fundamento Neurofisiológico

Como já fundamentado, a motivação na criação das redes neurais artificiais fora elaborada baseada no funcionamento do cérebro humano. Desta forma uma base neurofisiológica se torna apropriada para o entendimento das analogias feitas as redes neurais artificiais.

FREEMAN & SKAPURA, referindo-se à neurofisiologia alertam: “[...] *Bear in mind that the following discussion is a simple overview; the subject of neurofisiology is vastly more complicated than is the picture painted here*” (1991, p. 8).

De acordo com HAYKIN (2001), o sistema nervoso do ser humano pode ser dividido em três etapas básicas como demonstrado na Fig. 8. No fluxo proveniente da esquerda para a direita tem-se a alimentação, e da direita para a esquerda a retro-alimentação.

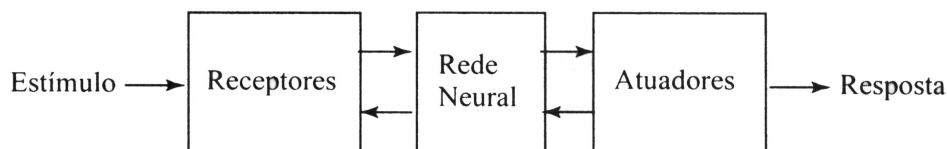


FIGURA 8: Esquema Padrão de uma Rede Neural

Fonte: HAYKIN 2001, p.32

HAYKIN (2001), expõe o funcionamento básico do sistema de redes neurais com início na conversão dos estímulos do corpo humano, pelos receptores, em um conjunto de sinais elétricos para esta rede neural (cérebro), que por sua vez efetua o

processamento, passando então para a área de ativação que apresenta um resultado do processamento efetuado pela rede como saída do sistema.

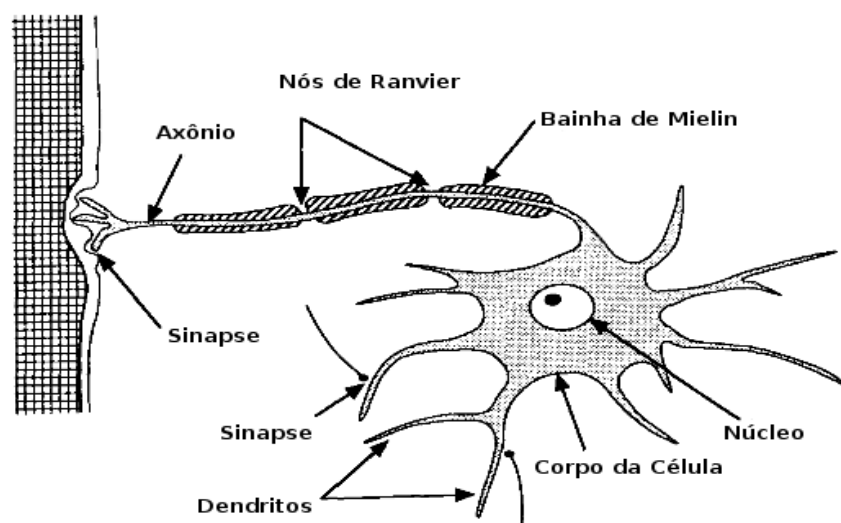


FIGURA 9: Morfologia de um Neurônio

Fonte: FREEMAN & SKAPURA 1991, p. 8 (adaptado pelo autor)

Para o entendimento de redes neurais é necessário também a definição do elemento estrutural para tal rede, o neurônio. FREEMAN & SKAPURA (1991) conforme a Fig. 9 demonstrando os principais componentes de uma célula nervosa (neurônio).

HAYKIN, define sinapses como: “As sinapses são unidades estruturais e funcionais elementares que mediam as interações entre neurônios [...] Nas descrições tradicionais da organização neural, assume-se que uma sinapse é uma conexão simples que pode impor ao neurônio receptivo excitação ou inibição, mas não ambas” (2001, p. 32). Ainda sobre sinapses, FREEMAN & SKAPURA definem:

Let's take a brief look at the activity that occurs at the connection between two neurons called the synaptic junction or synapse. Communication between neurons occurs as a result of the release by the presynaptic cell of substances called neurotransmitters, and of the subsequent absorption of these substances by the postsynaptic cell. [...] When the action potential arrives at the presynaptic membrane, changes in the permeability of the membrane cause an influx of calcium ions. These ions cause the vesicles containing the neurotransmitters to fuse with the presynaptic membrane and to release their neurotransmitters into the synaptic cleft.

The neurotransmitters diffuse across the junction and join to the postsynaptic membrane at certain receptor sites. The chemical action at the receptor sites results in changes in the permeability of the postsynaptic membrane to certain ionic species. An influx of positive species into the cell will tend to depolarize the resting potential; this effect is excitatory. If negative ions enter, a hyperpolarization effect occurs; this effect is inhibitory. Both effects are

local effects that spread a short distance into the cell body and are summed at the axon hillock. If the sum is greater than a certain threshold, an action potential is generated. (1991, p. 11)

É com a utilização das sinapses que os neurônios podem ser interligados formando redes, FREEMAN & SKAPURA (1991), utiliza a Fig. 10 para demonstração de alguns exemplos, onde a Fig. 10 (a) e (b) ilustram convergência (um neurônio recebe sinais de vários outros neurônios) e divergência (um neurônio envia sinais para vários outros neurônios) em redes neurais. *Feedback* pode ser notado na Fig. 10 (b), (c) e (d), onde a rede realimenta um neurônio formando um ciclo.

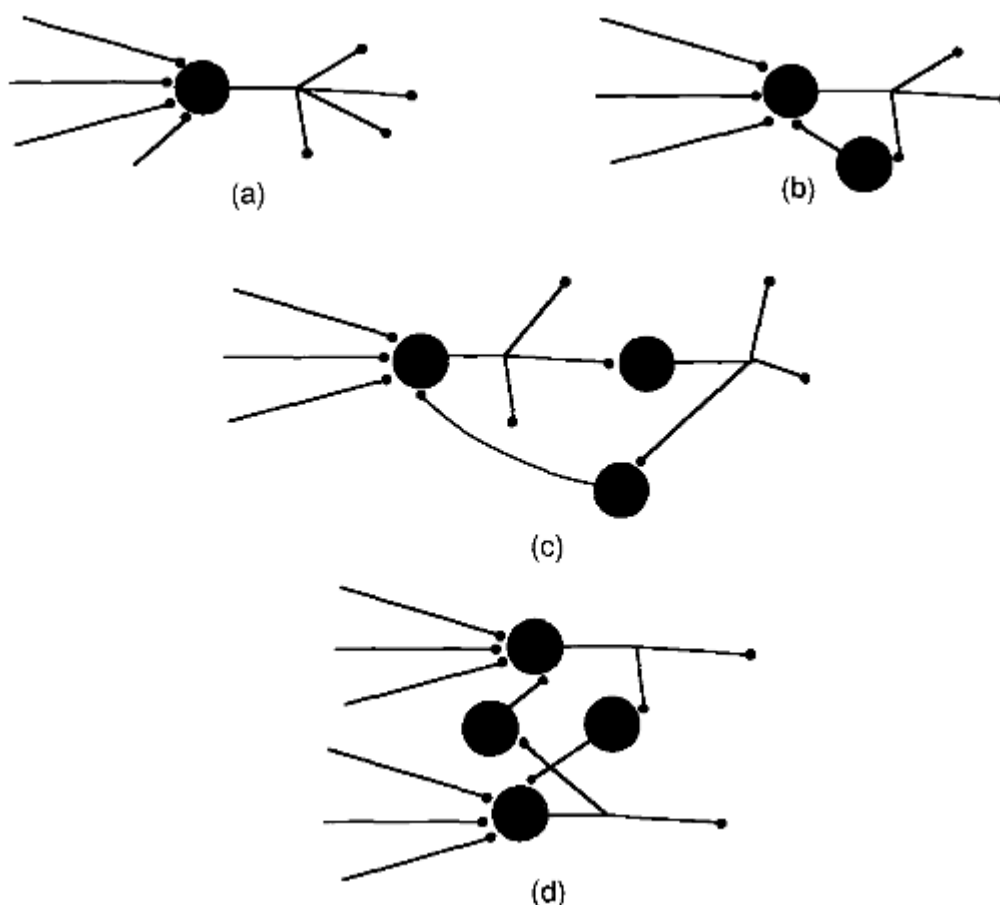


FIGURA 10: Vários modelos de neurônios interconectados
Fonte: Freeman 1991, p. 13

2.4.2 Introdução a Redes Neurais Artificiais

De forma geral, KRÖSE & VAN DER SMAGT define o processamento de cada unidade de uma rede neural (neurônio) como: “*Each unit performs a relatively simple job: receive input from neighbours or external sources and use this to compute an*

output signal which is propagated to other units.” (1996, p. 33)

HAYKIN (2001) expõe a mesma idéia de forma simples, apresentando a Fig. 11. que basicamente apresenta o esquema onde o componente central (soma), efetua o processamento dos valores recebidos, para através de uma função de saída apresentar um resultado.

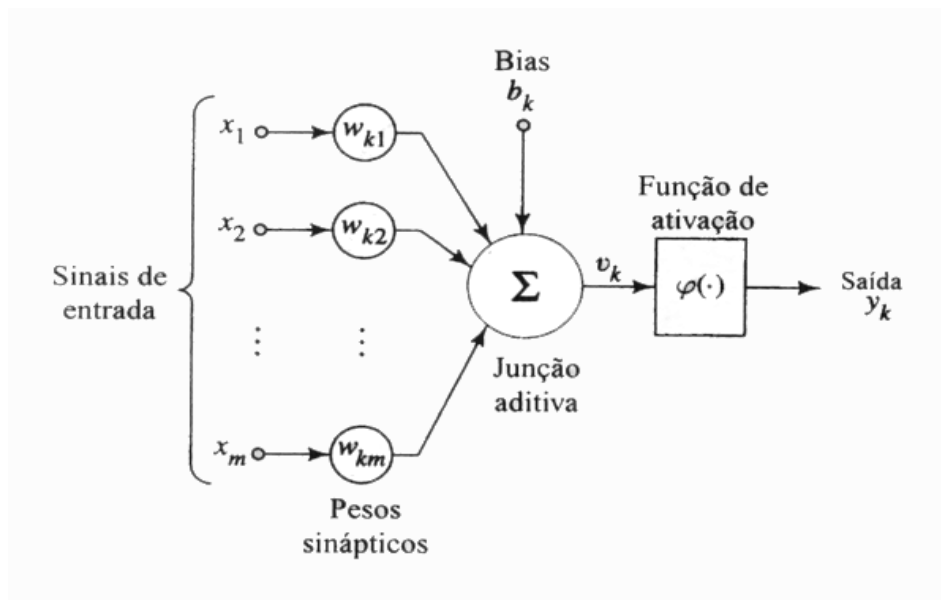


FIGURA 11: Funcionamento de um neurônio artificial

Fonte: HAYKIN 2001, p. 36

Através da Fig. 11 pode-se notar três elementos básicos, que de acordo com HAYKIN (2001) são:

1. Conjunto de sinapses – cada uma caracterizada por um peso, especificamente o sinal x_j , na entrada da sinapse j , conectado ao neurônio k que é multiplicado pelo peso w_{kj} ;
2. Um somador – para computar as entradas multiplicadas pelos seus respectivos pesos (no caso este processo constitui uma combinação linear);
3. Função de Ativação – para limitar a amplitude da saída de um neurônio, sendo assim limita-se a saída do neurônio a um valor finito.

A Fig. 11 apresenta ainda um elemento externo nomeado *bias*, denotado por b_k , que de acordo com HAYKIN (2001) tem por função aumentar ou diminuir o valor a ser passado para a função de ativação.

Em termos matemáticos, FREEMAN & SKAPURA (1991) e HAYKIN (2001)

apresentam a equação 26, que efetua a soma de todas as entradas a um neurônio multiplicada por seu peso sináptico.

$$u_k = \sum_{j=1}^m w_{kj} x_j \quad (26)$$

o valor extraído da equação é então utilizada na função de ativação, obtendo-se a saída do neurônio como pode-se notar na equação 27.

$$y_k = \varphi(u_k) \quad (27)$$

quando bias se faz presente este deve ser somado a u_k , e utilizado como parâmetro na função de ativação, o que de acordo com HAYKIN (2001) é expresso pela equação 28:

$$y_k = \varphi(u_k + b_k) \quad (28)$$

KRÖSE & VAN DER SMAGT (1996), assim como HAYKIN (2001), expõe como exemplo três funções de saída, conforme mostrado na Fig. 12.

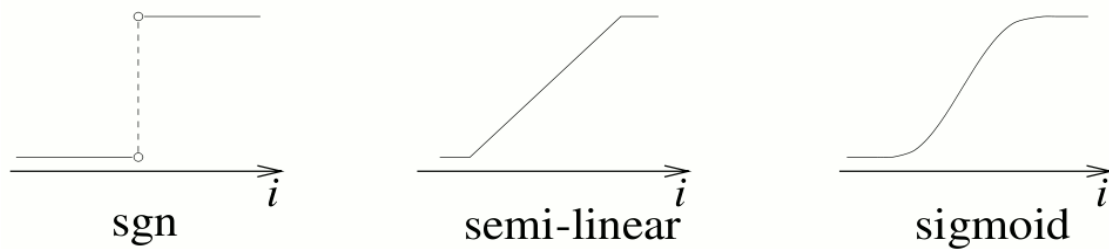


FIGURA 12: Exemplos de Funções de Ativação

Fonte: KRÖSE & VAN DER SMAGT (1996, p. 17)

A primeira das três funções expostas apresenta um limiar de corte, resultando somente valor um ou zero de acordo com o valor dado como entrada. A segunda apresenta um limiar de corte suave onde uma faixa é considerada e outra vale um ou zero. E para a função sigmoide, de acordo com HAYKIN (2001) tem-se a equação 29, que utiliza o parâmetro a para configurar o declive da função.

$$\varphi(v) = \frac{1}{1 + \exp^{-av}} \quad (29)$$

2.4.3 Topologia de Redes Neurais Artificiais

KRÖSE & VAN DER SMAGT (1996), dividem as topologias de redes neurais artificiais em dois grupos, o primeiro grupo, de redes alimentadas adiante (feed-forward) e o segundo grupo das recorrentes. As alimentadas adiante apresentam a característica do seu fluxo de dados fluir em um sentido único (no caso da entrada para a saída), onde nenhuma atividade de *feedback* é executada. O segundo grupo pertence então as redes que executam a retro-alimentação, isto é, os neurônios alimentam outros neurônios da mesma camada ou camadas anteriores, este ciclo de alimentação é executado então até que ocorra a estabilização da rede.

HAYKIN (2001) separa ainda o primeiro grupo citado por KRÖSE & VAN DER SMAGT (1996) em dois: de camada única e de múltiplas camadas. Nas que apresentam camada única tem-se somente a camada de entrada que projetam seus valores na camada de saída, os quais efetuam a computação no processo. E a topologia contendo múltiplas camadas distingue-se das de camada única por apresentar uma ou mais camadas intermediárias, ou escondidas, que também são camadas de neurônios que efetuam computações. A utilização de múltiplas camadas pode ser aplicado a uma grande quantidade de dados na camada de entrada, de forma que , estatísticas de ordem mais altas possam ser extraídas.

Tanto HAYKIN (2001) quanto KRÖSE & VAN DER SMAGT (1996) enfatizam que a utilização de redes neurais com topologia de retroalimentação geram ciclos, *loops*. Desta forma tanto o processamento por um resultado quanto à busca de forma automática dos pesos sinápticos de cada ligação entre os neurônios torna-se uma tarefa com custo computacional mais alto.

2.4.4 Aprendizado em Redes Neurais

De acordo com HAYKIN (2001), umas das propriedades primordiais para uma rede neural é a sua capacidade de aprender ou melhorar o resultado retornado através da aprendizagem. Basicamente, uma rede neural aprende ajustando seus pesos sinápticos e seus bias. Seguindo a mesma linha, LUGER (2004) define que o grande poder das redes neurais é justamente ter uma arquitetura de rede e um algoritmo de aprendizagem, desta

forma podem auxiliar na resolução de problemas sem serem especificamente programados para atender tal propósito.

Para HAYKIN (2001), o aprendizado em uma rede neural é iniciado quando a rede é estimulada pelo ambiente, que efetuam alterações em seus parâmetros, que por sua vez faz com que a rede responda de uma maneira nova ao ambiente onde está inscrita. Para efetuar esta operação tem-se os algoritmos de aprendizagem, que diferem entre si somente na forma como os pesos da rede são ajustados (HAYKIN 2001).

HAYKIN (2001), FREEMAN & SKAPURA (1991), KRÖSE & VAN DER SMAGT (1996) e LUGER (2004) citam a existência de dois grupos de algoritmos para treinamento de redes neurais, algoritmos de aprendizagem que necessitam de supervisão e algoritmos cuja supervisão não se faz necessária. Outros termos podem ser utilizados na literatura, como por exemplo “aprendizado por exemplo” citado por FREEMAN & SKAPURA (1991), referenciando-se à aprendizagem supervisionada ou ainda aprendizagem com e sem professor citado por HAYKIN (2001) referenciando a aprendizagem que necessita de supervisão e aprendizagem que não se faz necessária à supervisão, respectivamente.

A característica citada por LUGER (2001), no primeiro parágrafo deste item, é tomada como base para FREEMAN & SKAPURA (1991) explicar o funcionamento do modelo de redes que aprendem de forma supervisionada, onde são tomados exemplos para calibrar os pesos sinápticos e assim se adaptar ao ambiente onde a rede esteja operando. Para que os pesos sinápticos possam ser ajustados é preciso mensurar o erro causado por um retorno da rede para que as ligações sinápticas corretas possam ser ajustadas proporcionalmente a sua parcela de culpa no erro resultante. O objetivo, então, é efetuar iterações no treinamento para que o erro causado pela rede em sua resposta seja o menor possível. E de acordo com HAYKIN (2001), este método resulta na regra de aprendizagem denominada Regra Delta, criada por Widrow e Hoff em 1960.

Conforme HAYKIN (2001), um fator importante também na configuração de uma rede neural artificial é a taxa de aprendizagem, representada comumente por η , que de acordo com FREEMAN & SKAPURA (1991), é ideal que seja um número pequeno para que seja encontrado um ponto ótimo. Os pesos da rede são reajustados conforme uma taxa de aprendizagem, para que a correção seja feita de forma suave para evitar que

se passe do ponto ótimo. Como é exposto por FREEMAN & SKAPURA:

Selection of a value for the learning rate parameter, η , has a significant effect on the network performance. Usually, η must be a small number—on the order of 0.05 to 0.25 - to ensure that the network will settle to a solution. A small value of η means that the network will have to make a large number of iterations, but that is the price to be paid. It is often possible to increase the size of η as learning proceeds. Increasing η as the network error decreases will often help to speed convergence by increasing the step size as the error reaches a minimum, but the network may bounce around too far from the actual minimum value if η gets too large. (1991 p. 105)

Para cada caso uma taxa de erro é calculada. Este erro pode ser representado na forma de um hiperplano, assim a meta definida para o algoritmo é atingir o ponto mais fundo do hiperplano, onde o erro é menor.

LUGER (2004), e HAYKIN (2001) definem que a aprendizagem sem a necessidade de uma supervisão acontece de maneira independente, isto é, não se tem a definição do que é certo ou errado. RUSSELL & NORVIG esclarece este modelo de aprendizagem apresentando um exemplo.

O problema da aprendizagem não-supervisionada envolve a aprendizagem de padrões na entrada, quando não são fornecidos valores de saída específicos. Por exemplo, um agente de taxi poderia desenvolver gradualmente um conceito de “dias de tráfego bom” e “dias de tráfego ruim” sem jamais ter recebido exemplos identificando cada um deles. Um agente de aprendizagem puramente não supervisionado não pode aprender o que fazer, porque não tem nenhuma informação sobre o que constitui uma ação correta ou estado desejável. (2002, p. 630)

HAYKIN (2001) ainda organiza o grupo dos algoritmos de aprendizagem que não apresentam a necessidade de supervisão em dois subgrupos, os de aprendizagem por reforço e os de aprendizagem competitiva. LUGER (2004) define que o aprendizado por reforço se dá reforçando conexões sinápticas ativadas afim de classificar as entradas em grupos distintos. Este reforço se dá considerando unicamente a entrada e saída da rede. E o outro grupo, de aprendizagem competitiva, de acordo com HAYKIN (2001), a aprendizagem competitiva apresenta uma topologia de forma que a camada de neurônios competitivos tenha seus neurônios totalmente ligados entre si com conexões inibitórias fazendo com que o neurônio com maior valor resultante desligue os demais e se torne o vencedor. Ainda de acordo com HAYKIN (2001), a estrutura competitiva de redes neurais em sua forma mais simples opera com a estratégia “o vencedor leva tudo”.

Levando em consideração as definições apresetadas pelos autores, pode-se notar

que as redes onde a aprendizagem não é necessariamente supervisionada estão intimamente ligadas à tarefa de classificação, enquanto as supervisionadas podem ser utilizadas para aproximação de funções e detecção de padrões.

2.4.4.1 Aprendizado Hebbiano

O aprendizado de Hebb é baseado sob o livro *The Organization of Behavior*, e o nome permaneceu em homenagem ao seu autor. Citando Hebb:

Quando um axônio da célula A está perto o suficiente para excitar uma célula B e participa do seu disparo repetida ou persistentemente, então algum processo de crescimento ou modificação metabólica acontece em uma das células ou em ambas, de tal forma que a eficiência de A como uma das células disparadas por B é aumentada. (Hebb 1949, p. 62 apud Haykin, p. 80)

O método de Hebb tem sido amplamente utilizado em redes neurais artificiais, como comenta LUGER:

O aprendizado hebbiano tem sido usado numa série de arquiteturas de rede. Ele é usado tanto em aprendizado supervisionado como no modo não supervisionado. O efeito do reforço da conexão entre dois neurônios, quando um contribui para o disparo do outro, pode ser simulado matematicamente ajustando-se o peso nesta conexão por uma constante multiplicada pelo sinal do produto de seus valores de saída. (2004, p. 415)

FREEMAN & SKAPURA (1991) ilustra o aprendizado de Hebb apresentando um exemplo de arquitetura de rede como apresentada na Fig. 13, onde C causa o disparo de B, e é suposto ainda que enquanto B dispara, a ligação sináptica A-B está também ativa (auxilia no disparo de B). Neste caso o valor da conexão de A-B e C-B é incrementado. Porém, inicialmente A não causaria o disparo de B, mas depois de algumas iterações A poderia se fortalecer, permitindo que somente ele fizesse o disparo de B.

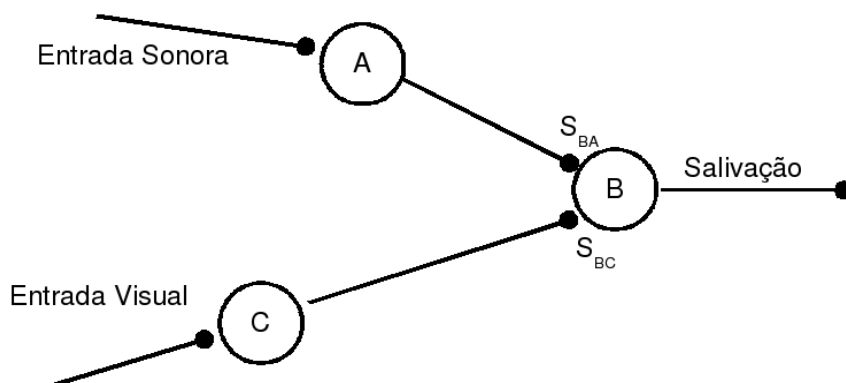


FIGURA 13: Aprendizagem por reforço

Fonte: FREEMAN & SKAPURA (1991, p.16) (adaptado pelo autor)

De forma simples, o aprendizado Hebbiano, conforme LUGER 2004, pode ser descrito matematicamente da seguinte forma, para dois neurônios (i e j) onde a saída de i é a entrada de j :

$$\Delta W = c(O_i * O_j) \quad (30)$$

A equação c representa a taxa de aprendizagem, O_i é o retorno do neurônio i e O_j é o retorno do neurônio j .

Da mesma forma, HAYKIN (2001) apresenta a equação abaixo para fortalecimento das conexões, e refere-se a ela como a regra do produto das atividades.

$$\Delta w_{kj}(n) = \eta(y_k(n) x_j(n)) \quad (31)$$

HAYKIN (2001) apresenta um problema resultante da aplicação do método acima descrito quando chega-se à saturação da conexão sináptica, assim nenhuma informação é armazenada depois de atingir tal ponto. Desta forma foi introduzida em 1977, por Sejnowski, a hipótese da covariância, onde os valores pré e pós sinápticos são substituídos pelo desvio de seus valores em relação a sua média em um certo espaço de tempo, ainda seguindo o princípio de Hebb do fortalecimento das conexões sinápticas. Abaixo é apresentada a equação abaixo para modelagem matemática da hipótese de covariância.

$$\Delta w_{kj}(n) = \eta(x_j - \bar{x})(y_k - \bar{y}) \quad (32)$$

2.4.4.2 Treinamento de Perceptrons

ROSENBLATT (1958) concebeu um algoritmo para treinamento de redes de *perceptrons* de camada única. De acordo com LUGER (2004), a entrada de dados numa rede de *perceptrons* de camada única é feita através dos valores 1 e -1. Os pesos sinápticos são valores reais e o nível de ativação de um *perceptron* é dado pela soma dos valores de entrada ponderados.

A função de ativação de perceptrons é dada pela função de limiar abrupto simples em que ativações acima de um determinado valor resultam 1 e abaixo resultam -1, o que LUGER (2004) apresenta como:

$$\begin{aligned} &1 \text{ se } \sum w_i x_i \geq t \\ &-1 \text{ se } \sum w_i x_i < t \end{aligned} \quad (33)$$

O cálculo de ajuste se dá pela equação:

$$\Delta w_i = c(d - \text{sgn}(\sum w_i x_i)) \quad (34)$$

Desta forma depois de efetuar uma interação de treinamento o erro é calculado, baseando-se sob o resultado obtido e o resultado correto, desta forma são modificados os pesos das ligações sinápticas do perceptron de forma a reduzir o erro gerado. A equação exige a entrada de alguns valores, onde **d** representa o valor desejado como retorno da rede; **c** representa a constante de taxa de aprendizagem; **sgn** é o valor de saída do perceptron e **w** o valor de entrada na i-ésima posição.

De acordo com LUGER (2004), os resultados possíveis para esta equação será 0, 2 ou -2. Se o retorno é 0 significa que o desejado e o obtido como retorno da rede são iguais, desta forma não é preciso efetuar nenhuma operação. Caso o valor de saída seja -1 e obtido fora 1, é incrementado os pesos da i-ésima linha em $2cx_i$. O mesmo acontece quando o desejado fora 1 e o obtido foi -1.

De acordo com FREEMAN & SKAPURA (1991), o trabalho de ROSENBLATT (1958), também resultou na prova de um importante artefato que é o teorema de convergência de um *perceptron*, que em síntese prova a possibilidade de que se existe algo que pode ser “aprendido” por um *perceptron*, uma quantidade limitada de ciclos pode ser executada para treinar o *perceptron*.

O fato perturbante, explorado por MINSKY & PAPERT (1969), é que os *perceptrons* efetuam separações lineares. Assim problemas não lineares como é o caso do **ou exclusivo**, como é comentado por FREEMAN & SKAPURA (1991), LUGER (2004) e HAYKIN (2001), poderia ser resolvido somente com uma rede de múltiplas camadas de *perceptrons*. No entanto o algoritmo que efetua o treinamento da rede já não seria mais eficaz necessitando então uma reformulação do algoritmo para ajuste de pesos, o que só fora possível só anos depois. A Fig. 14 apresenta o problema do **E lógico, Ou lógico e Ou exclusivo**.

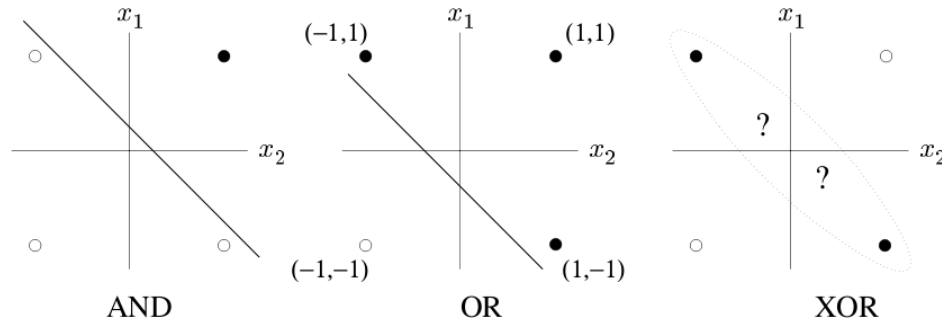


FIGURA 14: Representação geométrica do espaço de entrada

Fonte: KRÖSE & VAN DER SMAGT (1996, p. 29)

2.4.4.3 Regra Delta

Proposta por Widrow-Hoff em 1960, tem como objetivo efetuar a correção dos pesos sinápticos baseado sobre o erro causado pelo resultado apresentado. Conforme HAYKIN (2001), o treinamento ocorre de forma a apresentar os dados de entrada em um dado instante de tempo n $\mathbf{x}(n)$, a saída desejada do neurônio k $\mathbf{d}_k(n)$ e a saída recebida do neurônio k $\mathbf{y}_k(n)$. Desta forma a medida do erro $e_k(n)$ é dada por:

$$e_k(n) = d_k(n) - y_k(n) \quad (35)$$

Citando HAYKIN tem-se:

O sinal de erro $e_k(n)$ aciona um mecanismo de controle, cujo propósito é aplicar uma sequência de ajustes corretivos aos pesos sinápticos do neurônio k . Os ajustes corretivos são projetados para aproximar passo a passo o sinal de saída $y_k(n)$ da resposta desejada $d_k(n)$. Este objetivo é alcançado minimizando-se uma função de custo ou índice de desempenho, $\xi(n)$, definido em termos do sinal de erro $e_k(n)$ como:

$$\xi(n) = \frac{1}{2} e_k^2(n)$$

Com isso, $\xi(n)$ é o valor instantâneo da energia do erro. Os ajustes passo a passo dos pesos sinápticos do neurônio k continuam até o sistema atingir um estado estável (i.e., os pesos sinápticos estão essencialmente estabilizados). Neste ponto o processo é encerrado. (2001, p. 77)

Com isso, o ajuste do peso sináptico w_{kj} que acontece em um neurônio k estimulado por um elemento $x_j(n)$ do vetor de sinal $\mathbf{x}(n)$ em um tempo n a uma taxa de aprendizagem η é representado por:

$$\Delta w_{kj}(n) = \eta e_k(n) x_j(n) \quad (36)$$

HAYKIN exprime de forma simples a regra delta como: “O ajuste feito em um peso sináptico de um neurônio é proporcional ao produto do sinal de erro pelo sinal de

entrada da sinapse em questão.” (2001, p. 78).

Tendo calculado o ajuste a ser efetuado, o valor do peso sináptico ajustado, de acordo com HAYKIN (2001), é determinado por

$$w_{kj}(n+1) = w_{kj}(n) + \Delta w_{kj}(n) \quad (37)$$

2.4.4.4 Algoritmo de Retropropagação

De acordo com MICHELI-TZANAKOU (2000), o algoritmo de retropropagação, caracteriza-se por ser um método de aprendizagem onde, como a regra delta, a taxa de erro causado por um neurônio na camada de saída é comparado com o valor desejado e retropropagado seu erro para os neurônios das camadas intermediárias, ajustando assim os pesos das ligações sinápticas existentes entre os neurônios.

HAYKIN (2001) expõe que o sinal de erro calculado se dá utilizando a equação de medida de erro da regra delta. Sabendo que os únicos neurônios “visíveis” para computação de erro de uma rede neural são os neurônios de saída, a energia total do erro, é dada por

$$\xi(n) = \frac{1}{2} \sum_{j \in C} e_j^r(n) \quad (38)$$

desta forma o erro médio resultante da apresentação de cada exemplo do conjunto de treinamento é estabelecido pela média dos erros totais pelo número de padrões, o que pode ser expresso por

$$\xi_{med} = \frac{1}{N} \sum_{n=1}^N \xi(n) \quad (39)$$

A correção do peso sináptico acontece seguindo a equação:

$$\Delta w_{ji}(n) = -\eta \frac{\partial \xi(n)}{\partial w_{ij}(n)} \quad (40)$$

η é a taxa de aprendizagem do algoritmo de retropropagação, ainda de acordo com HAYKIN (2001), o sinal negativo indica a decida do gradiente no espaço de pesos, e

$$\Delta w_{ji}(n) = \eta \delta_j(n) y_i(n) \quad (41)$$

sendo que

$$\delta_j(n) = \frac{\partial \xi(n)}{\partial v_j(n)} = -\frac{\partial \xi(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} = e_j(n) \varphi_j'(v_j(n)) \quad (42)$$

desta forma indica as correções necessárias para os pesos sinápticos.

$$v_j(n) = \sum_{i=0}^m w_{ij}(n) y_i(n) \quad (43)$$

onde m é o total de entradas, com exceção de valores provenientes de bias, para o neurônio j . Assim, $v_j(n)$ é utilizado junto a função de ativação para calcular a saída do neurônio j .

De acordo com a definição de HAYKIN (2001) e MICHELI-TZANAKOU (2000), pode-se encaixar tal algoritmo como um método de aprendizagem supervisionado, uma vez que utiliza padrões de entrada e faz a correção dos pesos baseado em um método para medir o erro.

2.4.5 Redes MLP (Multilayer Perceptron)

HAYKIN (2001) define o *perceptron* como a forma mais simples da rede neural usada para classificação de padrões dito linearmente separáveis. A rede consiste basicamente em uma estrutura com pesos sinápticos e bias balanceáveis.

A estrutura de rede baseada em perceptrons bem como um meio para ajuste dos pesos, fora desenvolvido em 1958, como já fundamentado por Frank Rosenblatt, onde a idéia apresentada para o ajuste de pesos sinápticos acontecia efetuando a aplicação do teorema de convergência de perceptrons. Inicialmente, os pesos sinápticos eram aleatórios e ajustados no decorrer do processo de treinamento (FREEMAN & SKAPURA, 1991). A utilização de um único neurônio na estrutura da rede poderia separar linearmente duas classes, por outro lado criando uma estrutura onde mais de um neurônio é utilizada, tem-se a separação em mais de duas classes, mas precisam ainda ser linearmente separáveis (HAYKIN, 2001). A estrutura tem como princípio fazer uso de uma função de ativação com um limiar de corte.

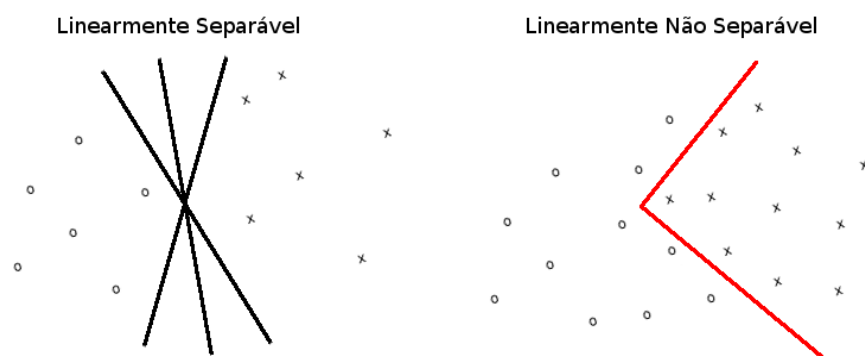


FIGURA 15: Separabilidade linear

O *perceptron* é criado sob um neurônio linear, mas pode também ser criado a partir de um modelo de neurônio não linear, como fundamentado no modelo sugerido por McCulloch e Pitts, onde é feito uso de uma função de ativação tipo sigmoide (HAYKIN 2001). A utilização de funções contínuas, como citado por HAYKIN (2001), de acordo com LUGER (2004), aproximam a função de limiarização criada por um corte abrupto. Assim a taxa de erro utilizada no algoritmo de aprendizagem pode ser calculado de forma mais precisa. Outra propriedade também interessante, citado por LUGER (2004), é de que a função contínua da sigmoide tem uma área de transição rápida, fazendo com que a maioria dos valores fiquem próximos as regiões mínimas e máximas.

As redes *Multilayer Perceptron* são uma generalização da rede de *perceptron* de camada única, com arquitetura de rede alimentada adiante e múltiplas camadas, como pode-se observar na Fig. 16, o que dá o poder de separação de padrões não linearmente separáveis (HAYKIN, 2001). Tem sido constantemente aplicado em diversos trabalhos para a resolução de problemas, de acordo com ROWLEY, BALUJA & KANADE (1998) e IMRIE & DURUCAN (2000).

Confome HAYKIN (2001), os únicos neurônios que efetuam algum processamento computacional estão presentes nas camada intermediárias, também denominadas de camadas ocultas, e na camada de saída. A primeira camada oculta é alimentada pelas entradas, e são denominados de nós sensoriais. A propagação dos dados da primeira camada oculta em direção da camada de saída é feita até atingí-la. Cada neurônio das camadas ocultas ou das camadas de saída são desenvolvidos para efetuar o cálculo da função de ativação que é propagado para a próxima camada ou no

caso da camada de saída apresentada como retorno do processamento efetuado pela rede e o cálculo da estimativa do vetor gradiente que é necessário para o algoritmo de retropropagação efetuar o ajuste dos pesos das conexões sinápticas.

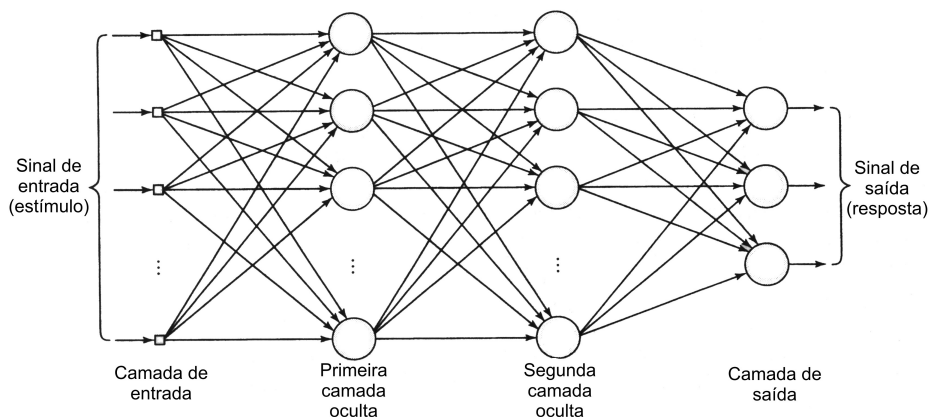


FIGURA 16: Estrutura de uma rede de perceptrons de múltiplas camadas

Fonte: HAYKIN (2001, p. 186)

Esta classe de rede é treinada de forma supervisionada com o algoritmo de retropropagação do erro, ou popularmente chamado *backpropagation* (HAYKIN, 2001).

As características básicas apresentadas para uma MLP, de acordo com HAYKIN (2001) são:

1. função de ativação não linear, o que fornece a característica de suavidade no retorno da função de ativação, diferentemente do modelo rígido sugerido por Rosenblatt, como exemplo tem-se a função sigmoidal;
2. a rede possui uma ou mais camadas escondidas, o que dá o poder da extração de características cada vez mais complexas dos vetores de valores de entrada;
3. a rede apresenta grande quantidade de conexões determinadas pela suas sinapses.

2.4.6 Aproximação de Funções / Reconhecimento de Padrões

Uma das tarefas executadas com grande sucesso pelas redes neurais é a aproximação de função, o que permite o mapeamento de uma função $f(x)$ onde entrada-saída podem ser não lineares (HAYKIN, 2001).

De acordo com HAYKIN (2001) pode-se aproximar a uma função desconhecida compensando a falta de conhecimento, fornecendo um conjunto de exemplos objetivando a criação de uma função $F(x)$ que apresente $\|F(x)-f(x)\|$ pequeno para todo

x .

HAYKIN (2001) denomina que um vetor de valores desejados (d_i) é apresentado em função de um vetor de dados de entrada (x_i) faz algumas considerações sobre a aproximação de função relevantes:

O problema de aproximação descrito aqui é um candidato perfeito para a aprendizagem supervisionada, com x_i desempenhando o papel de vetor de entrada e d_i desempenhando o papel da resposta desejada. Podemos então inverter esta questão e ver a aprendizagem supervisionada como um problema de aproximação. (HAYKIN, 2001, p. 94)

3 MATERIAL E MÉTODO

A utilização de wavelets como ferramenta para análise de séries temporais já é amplamente utilizada. Isto é possível constatar em BOTELLA, ROSA-HERRANZ & GINER (2003) e PERCIVAL (2000) também como um método extrator de características como já fundamentado. Desta forma a extração de características de uma série temporal pode ser utilizado como fonte de dados para efetuar uma aproximação de função através de uma rede neural de perceptrons de multiplas camadas alimentadas a diante.

Fica a dúvida quanto ao tipo de wavelet a ser empregada para que a extração de característica ocorra obtendo-se melhor desempenho computacional para aproximação de função sobre estas características. Para tanto a idéia é utilizar a análise de variância sobre testes repetitivos que aplicam diferentes tipos de wavelets podendo desta forma definir qual a melhor configuração wavelet que facilite a extração de características de forma a complementar a atividade de aproximação de função desempenhado pela rede neural.

Serão analisadas as seguintes wavelets: haar, para um teste controle sendo a mais simples das wavelets e que será utilizado para a comparação com os outros tipos de wavelets. Daubechies (DAUBECHIES, 1992), outra wavelet, também já utilizada no trabalho BOTELLA, ROSA-HERRANZ & GINER (2003) como ferramenta para análise de dados, são testada as ordens 3, 5, 7, 10, 15 e 20, Coiflet (DAUBECHIES, 1992), também aplicada em outros trabalhos de análise climática como pode ser visto em YNDESTAD (2006) são testadas as ordens 1, 2, 3, 4 e 5, Symlet (DAUBECHIES, 1992), que também é aplicada em trabalhos de análise de dados utilizando wavelets e são testadas as ordens 3, 5, 7, 10, 15 e 20, Chroma (DOVICCHI, 1999) nas ordens 4 e 6.

Baseado sobre a melhor wavelet definida no passo anterior será efetuado outra bateria de testes afim de definir qual o melhor janelamento e sobreposição da série e tempo despendido para trienamento/inferência por parte da rede neural. Desta forma, espera-se obter como resultado características semelhantes ou melhores quanto aos resultados apresentado, aplicando apenas redes neurais mas com um custo computacional menor em virtude da apresentação de dados decompostos para

treinamento e inferência da rede neural.

A mesma bateria de testes será realizada para resolução em nível 1, 2 e 3 para averiguar o comportamento do sistema preditivo quando decomposto em mais níveis.

Para a rede neural, serão utilizadas 3 camadas, sendo que as duas primeiras camadas utilizarão a função tagente sigmoidal para ativação e a camada de saída uma função linear. As duas primeiras camadas possuirão dez neurônios cada e a última somente um, que será o valor retornado pela rede. Para que o aprendizado se dê em um gradiente suave, é definido uma taxa de aprendizagem baixa (o que é aconselhado por FREEMAN & SKAPURA (1991) como fundamentado). Para o reajuste dos pesos será aplicado o algoritmo de retropropagação com a otimização de Levemberg e Markquart. A representação gráfica final da rede é exposta na Fig. 17, onde quadrados representam as entradas da rede, e os círculos cada um dos neurônios, também é exposto sobre cada camada seu nome e a função de ativação utilizada.

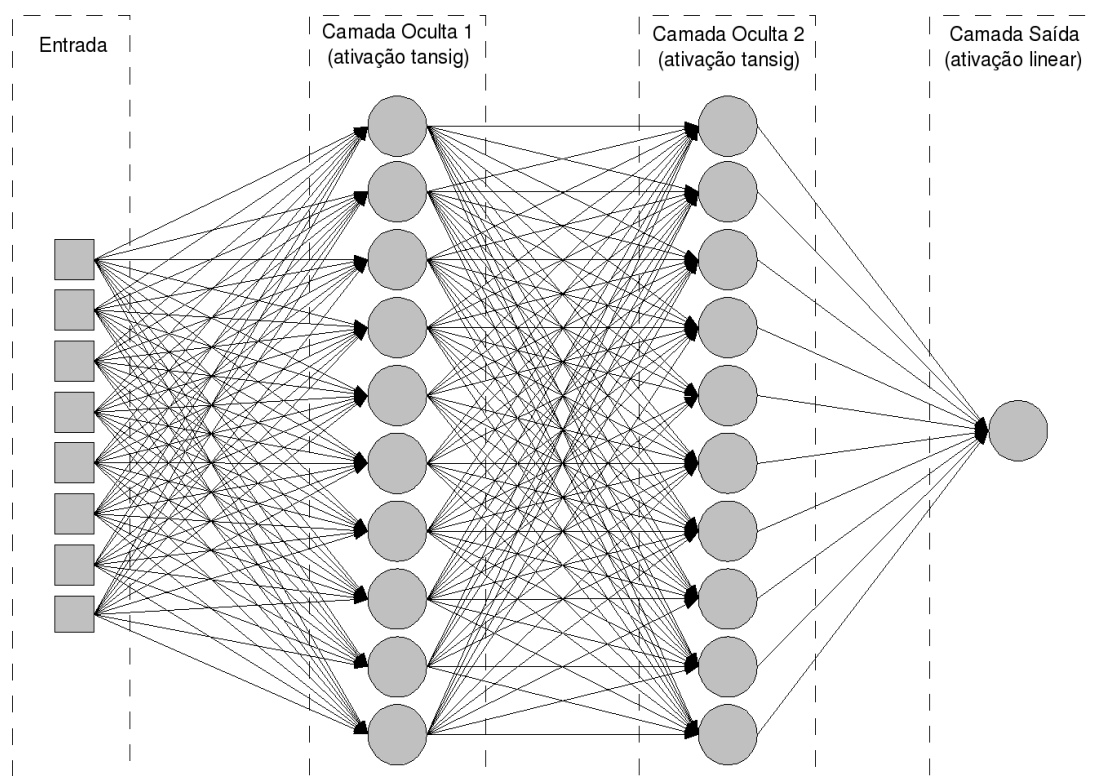


FIGURA 17: Estrutura da Rede Neural utilizada no modelo para testes com entrada de oito elementos.

Para o treinamento da rede neural será utilizado 75% dos dados da série temporal, a outra parte (25%) será utilizado para inferência, extraíndo o tempo e erro causado pelo

modelo.

A série temporal utilizada para validação do modelo são características históricas extraídas do Rio Itajaí Açu, lidos na cidade de Rio do Sul fornecidos pela Epagri. Os dados fornecidos compreendem a leitura de vazão do rio de fevereiro de 1977 a fevereiro de 2007 (sendo efetuada uma leitura a cada dia, totalizando 1.0531 leituras). É importante ressaltar que os trechos que comprometem o treinamento (trechos não lidos) foram removidos. A série de leituras efetuadas é exposta na Fig. 18.

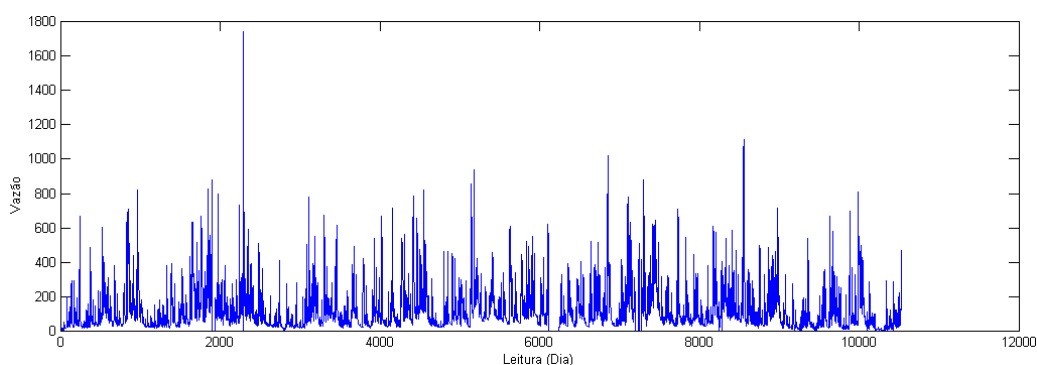


FIGURA 18: Série histórica das leituras de vazão do Rio Itajaí Açu, de fevereiro de 1970 a fevereiro de 2007, efetuadas na cidade de Rio do Sul.

Para efetuar a análise de forma padrão para todas as wavelets, será utilizado janelamento de dezesseis leituras e sobreposição de oito leituras. Isto significa que de oito em oito leituras serão lidas as próximas 16 para a extração de características e o décimo sétimo será utilizado para indicar o resultado esperado da rede. A Fig. 19 exemplifica a extração dos dezesseis elementos em um primeiro momento (de um a dezesseis) utilizando o décimo sétimo como parâmetro para o treinamento supervisionado indicando a resposta correta. Sobrepondo 8 elementos tem-se o segundo passo de nove até vinte e quatro com a resposta correta no vigésimo quinto elemento, que servirá como outro caso para o treinamento supervisionado. O processo é feito consecutivamente até o final da série utilizada para treinamento ou teste.

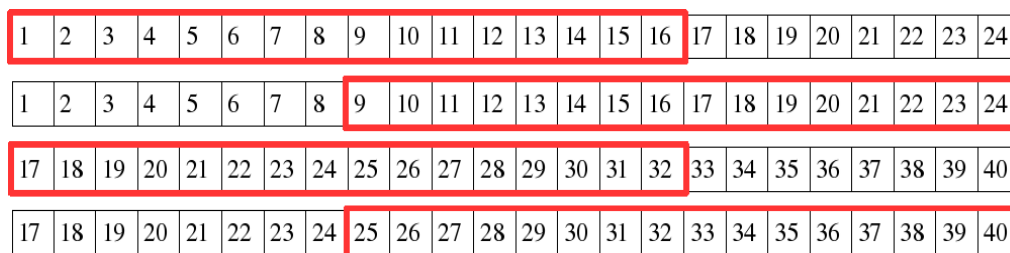


FIGURA 19: Sobreposição de janela na série temporal

Para fins comparativos, serão extraídos os erros dos métodos aplicados de duas formas, a primeira será extraído o número de acertos considerando uma margem de um desvio padrão da série temporal original para o simulado, considerando somente se está dentro da margem ou não. Caso esteja, será considerado certo, caso não esteja será considerado errado. Outra forma a ser aplicada será o erro quadrático, para saber qual o erro gerado em m^3/s , desta forma será englobado todo o erro gerado pela previsão.

Para os testes é utilizado o ambiente MatLab com a toolbox de redes neurais e wavelets. Para cada configuração de testes efetuados serão extraídos os erros, dados simulados e o tempo de execução do treinamento para posterior comparação. Os testes serão efetuados sobre uma máquina que apresenta 1024 Mb de RAM, utilizando processador intel centrino de 1,7 Ghz e sistema operacional Windows XP.

3.1 DESCRIÇÃO DOS ALGORITMOS

Cria-se uma função de suporte nomeada *prever*, cujo uso será feito nos outros algoritmos. Esta tem por objetivo a criação de uma rede neural, configuração, treinamento e inferência, retornando desta forma à rede treinada, a fração da série de dados utilizada para teste, o resultado simulado e o tempo aplicado em cada etapa do processo (da criação do conjunto de dados de treinamento a inferência dos dados na rede). Como parâmetros serão exigidos, o janelamento utilizado, sobreposição para os dados de treinamento, a wavelet utilizada para decomposição dos dados de treinamento, nível de decomposição, dados para treinamento, os limites para os dados de treinamento e teste, e a quantidade de épocas para treinamento, onde o valor padrão é 20.

```
[rede, correto, simulado, tempos] = função prever(janela, sobreposicao, wavelet,
decomposição, dados, limites, epocas = 20)
conjuntoTreinamento = {};
resultadoTreinamento = {};
conjuntoSimulação = {};
resultadoSimulação = {};
tempos[1] = agora;
para t=limites[1] passo sobreposicao até limites[2] - janela
    conjuntoTreinamento[t/sobreposicao] = transformacaoDiscreta(dados[t..t
+janela-1],wavelet,decomposição);
    resultadoTreinamento[t/sobreposicao] = dados[t+janela];
fimpara
tempos[1] = tempos[1] - agora;
tempos[2] = agora;
para t=limites[3] passo sobreposicao até limites[4]-janela
    conjuntoSimulacao[t/sobreposicao] = transformacaoDiscreta(dados[t..t
```

```

        +janela-1],wavelet,decomposiçao);
    resultadoSimulacao[t/sobreposicao] = dados[t+janela];
fimpara
tempos[2] = tempos[2] - agora;
rede = novaRedeMLP();
rede.epocasTreinamento = 20;
tempos[3] = agora;
treinarRede(rede,conjuntoTreinamento,resultadoTreinamento);
tempos[3] = tempos[3] - agora;
tempos[4] = agora;
simulado = simular(rede,conjuntoSimulacao);
tempos[4] = tempos[4] - agora;
correto = resultadoSimulacao;
fimfunção

```

Para a extração de dados estatísticos também é criado uma função nomeada *taxaErro*, que retorna um índice percentual de acerto e erro quadrático médio gerado, como parâmetros recebe-se as duas seqüências de dados e o janelamento utilizado para efetuar o treinamento e inferência.

```

[acertoPercentual, erroQuadraticoMedio] = função taxaErro(correto,simulado)
para i=1 até tamanho(correto)
    diferenca = correcto[i] - simulado[i];
    valorQuadratico = raizQuadrada(diferenca^2);
    erroQuadraticoMedio = erroQuadraticoMedio + valorQuadratico/tamanhoCorreto;
    se ( valorQuadratico <= desvioPadrao(correto))
        acertoPercentual = acertoPercentual + 1/tamanho(correto);
    fimse
fimpara
fimfunção

```

Os dois algoritmos tratados até o presente momento estão anexos ao trabalho no apêndice I.

Para definir a melhor wavelet é utilizado janelamento de 16 leituras, sobreposição de 8 e 75 repetições, assim tem-se o algoritmo que segue:

```

waveletCandidatas = {'haar', 'db3', 'db5', 'db7', 'db10', 'db15', 'db20','coif1',
'coif2', 'coif3', 'coif4', 'coif5', 'sym3', 'sym5', 'sym7', 'sym10', 'sym15',
'sym20', 'chma4', 'chma6','semwv'};
limites = {3000,9000,9000,10500};
dadosPercentuais = {};
dadosErroQuadratico = {};
para w=0 até tamanho(waveletCandidatas)
    para r=1 até 75
        [rede, correcto, simulado, tempos] = prever(16,8,waveletCandidatas[w],
            1,dados,limites);
        [percAcerto,erroQuadMed] = taxaErro(correto,simulado);
        dadosPercentuais[r] = percAcerto;
        dadosErroQuadratico[r] = erroQuadMed;
    fimpara
fimpara

```

O mesmo princípio é utilizado para a execução de outros dois testes: a combinação de janelamento / sobreposição, assim como para verificar o tempo despendido pelos processos em configurações onde quantidade de dados diferentes são tomados como fonte para a extração de características.

4 RESULTADOS

O resultados serão divididos em três etapas: a primeira é a comparação entre as wavelets para constatar qual a melhor wavelet para efetuar a tarefa de extração de características para o sinal desejado; a segunda será abordado qual o janelamento e sobreposição ideal dos dados para o modelo; a terceira será comentado o esforço computacional realizado no método e comparado com a aplicação redes neurais sem o pré-processamento wavelet.

O modelo criado para efetuar os testes segue o que foi definido no capítulo de material e meodo, na Fig. 20 pode-se verificar a existência do pré-processamento wavelet dos dados, desta forma as características extraídas dos dados são passados para a rede neural artificial e esta faz a aproximação baseada sobre o seu treinamento desempenhando assim seu papel previsivo.

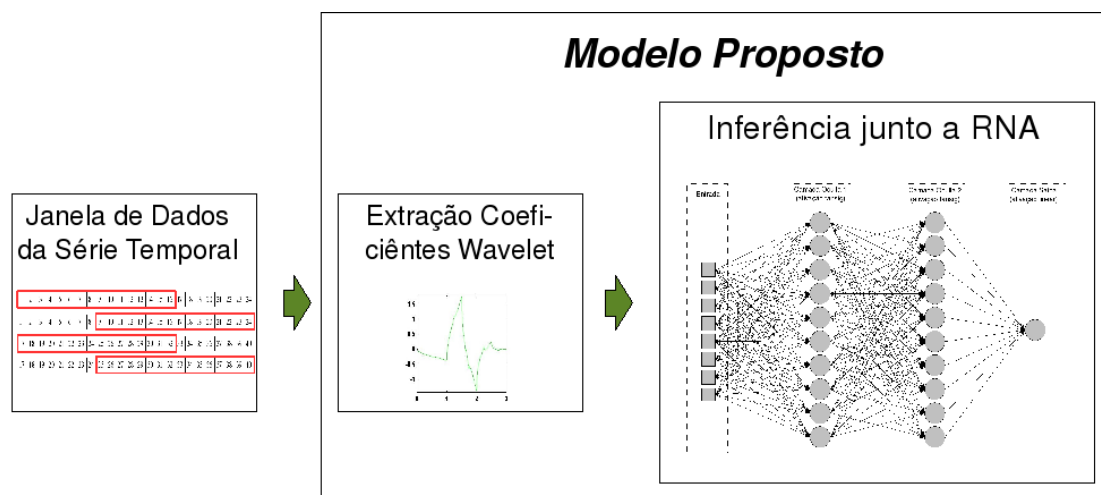


FIGURA 20: Funcionamento do modelo proposto (do janelamento de dados da série temporal a inferência por parte da Rede Neural Artificial).

4.1 COMPARAÇÃO ENTRE WAVELETS

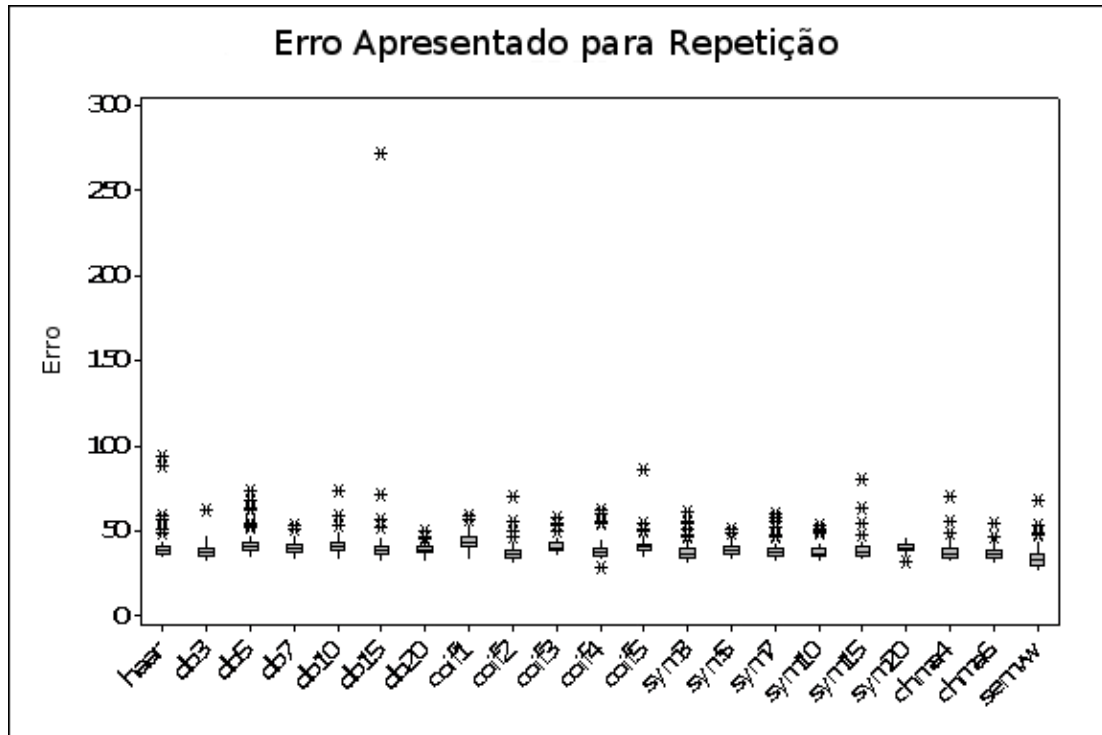


FIGURA 21: Diagrama de caixas com os erros quadráticos das 75 execuções agrupadas

Como já introduzido no capítulo anterior, as wavelets abordadas são Haar (haar), Daubechies nas ordens 3, 5, 7, 10, 15 e 20 (db3, db5, db7, db15 e db20); Coiflet nas ordens 1, 2, 3, 4 e 5 (coif1, coif2, coif3, coif4 e coif5); Sylet nas ordens 3, 5, 7, 10, 15 e 20 (sym3, sym5, sym7, sym10, sym15 e sym20) e Chroma nas ordens 4 e 6 (chma4 e chma6). São feitas 75 repetições para cada wavelet afim de apresentar um nível de significância alto quanto a definição do melhor método.

Pode-se constatar que, de acordo com a Fig. 21, foram removidos os testes que apresentavam taxas de erro quatro vezes superior a sua média, para que estes não afetem os resultados sendo eles casos excepcionais, resultando desta forma na Fig. 22.

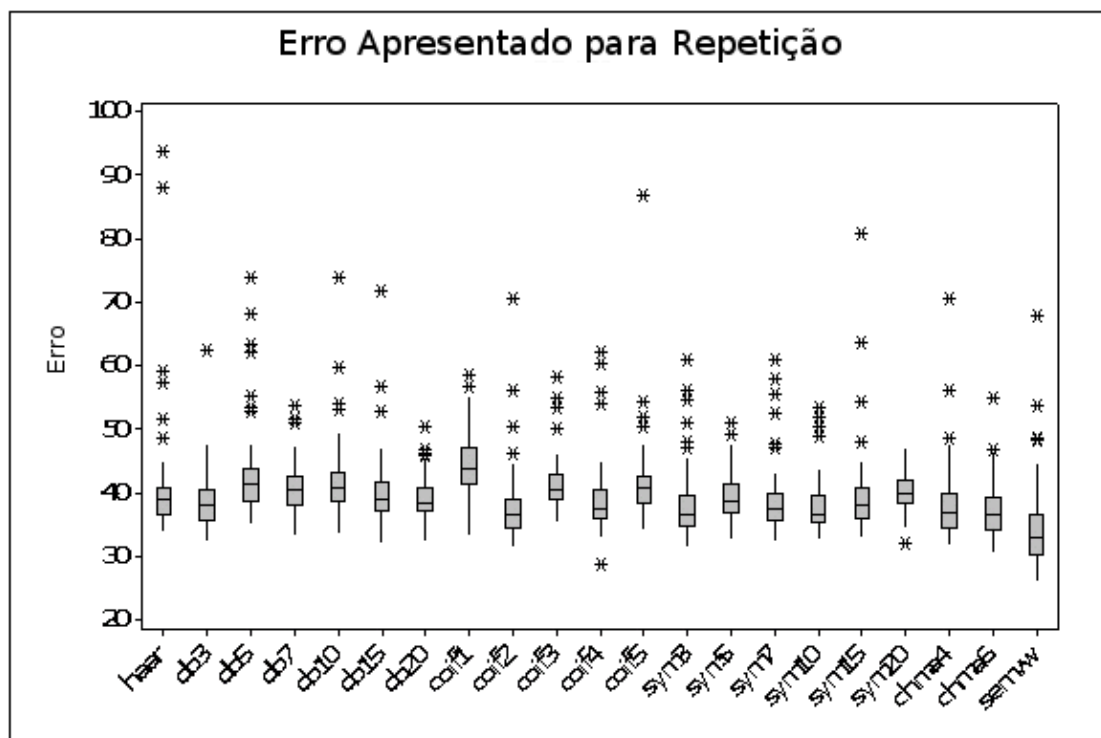


FIGURA 22: Diagrama de caixas com os erros quadráticos das execuções agrupadas sem outliers

Na Fig. 23 é apresentado o erro quadrático de cada uma das ondas aplicadas para a extração de características utilizadas pela rede neural para a atividade de aproximação de função.

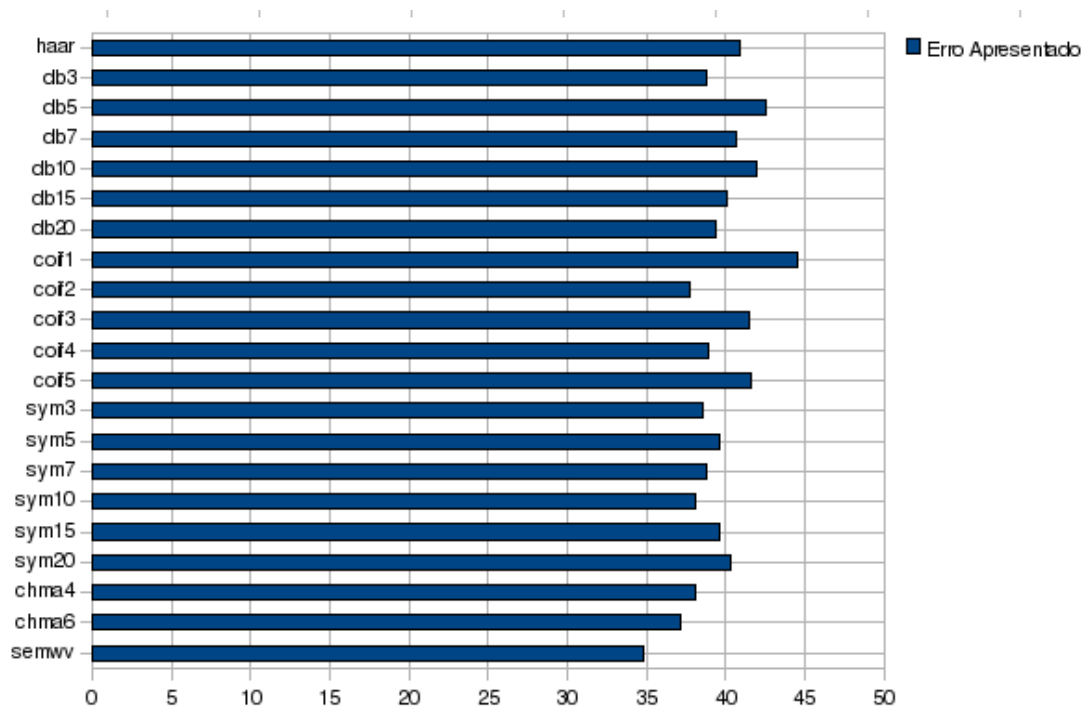


FIGURA 23: Erro médio apresentado em cada caso.

Sobre os valores resultantes da rede, também foram efetuados cálculos onde a previsão do valor dentro da margem de um desvio padrão é considerada certo, fora desta margem é considerado errado. O retorno deste processamento produz os dados apresentados na Fig. 24.

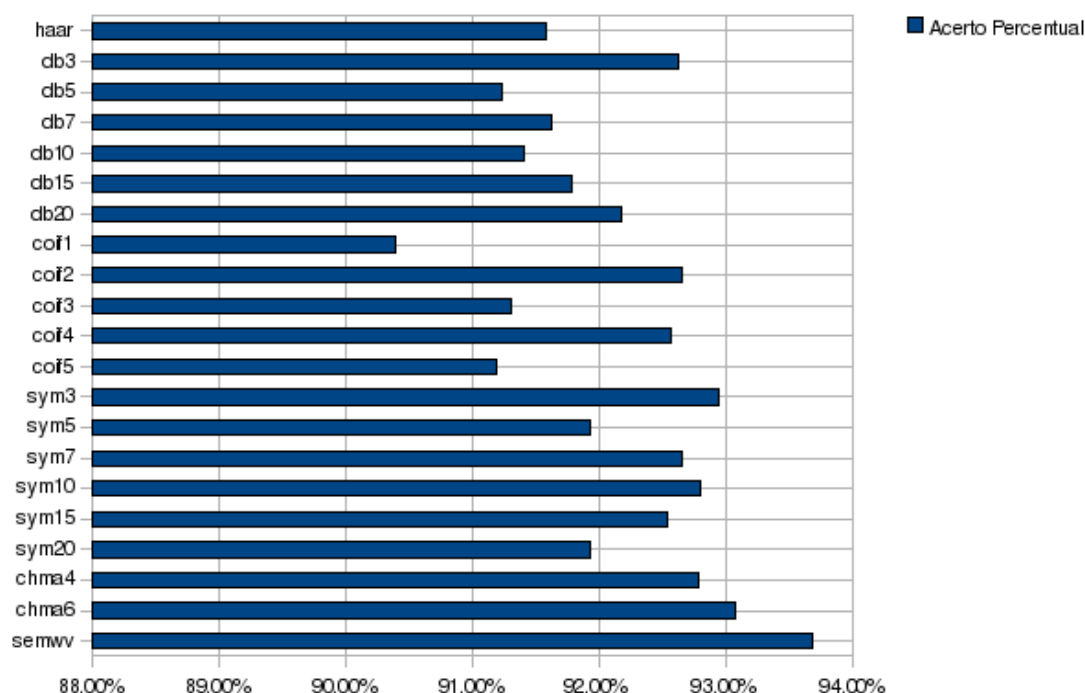


FIGURA 24: Acerto percentual médio apresentado para cada wavelet.

4.2 COMPARAÇÃO JANELAMENTO / SOBREPOSIÇÃO

Para testes de janelamento e sobreposição é medido o tempo (em segundos), erro quadrático e o erro percentual baseado sobre o mesmo processo de acerto do item abordado anteriormente (margem de um desvio padrão).

Para detalhar questões relativas ao tempo de processamento, este é dividido em 4 etapas: a primeira delas (ET1) faz referência à carga dos dados de treinamento de forma que seja armazenado em um vetor todos os conjuntos de valores resultantes do janelamento efetuado na série para efetuar o treinamento da rede neural; a segunda (ET2) diz respeito ao treinamento da rede neural (utilizando os dados provenientes da primeira etapa como parâmetros para treinamento); a terceira (ET3) carga dos dados para teste, que tem por objetivo a geração dos parâmetros para efetuar a inferência de teste e marcar seu índice de acerto e erro, segue o mesmo processo efetuado na primeira etapa; e a quarta etapa (ET4) que é a inferência dos dados na rede treinada, podendo

extrair o índice de erro do método.

A combinação de janela/sobreposição utilizada é definida na Fig. 25, onde J

	1	2	3	4	5
1	J 16 - S 1	J 16 - S 2	J 16 - S 4	J 16 - S 8	J 16 - S 16
2	J 32 - S 2	J 32 - S 4	J 32 - S 8	J 32 - S 16	J 32 - S 32
3	J 64 - S 4	J 64 - S 8	J 64 - S 16	J 64 - S 32	J 64 - S 64
4	J 128 - S 8	J 128 - S 16	J 128 - S 32	J 128 - S 64	J 128 - S 128
5	J 256 - S 16	J 256 - S 32	J 256 - S 64	J 256 - S 128	J 256 - S 256

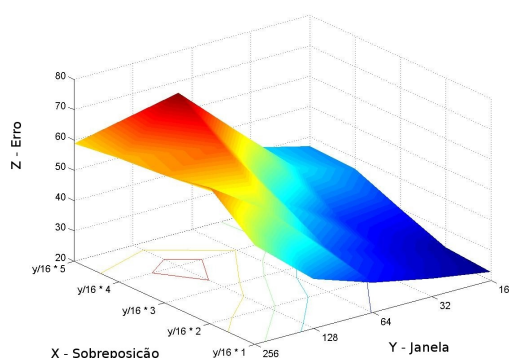
FIGURA 25: Combinação Janela / Intercalação utilizada

precede o número de elementos utilizado no processo de janelamento e S o número de dados sobrepostos no processo de janelamento da série. Desta forma para cada caso (utilizando somente redes neurais, utilizando chroma com uma decomposição, duas decomposições e três decomposições) serão aplicadas todas as vinte e cinco combinações, e extraídos seu tempo (em segundos) e erro em relação ao esperado.

4.2.1 Somente Rede Neural

A Fig. 26 apresenta o comportamento padrão do erro quadrático da rede neural para cada combinação proposta, sem a aplicação de wavelets para o pré-processamento dos dados janelados.

A Fig. 27 demonstra o comportamento da rede neural para com o acerto



y/x	y/16	2y/16	4y/16	8y/16	y
16	22.79	24.13	30.01	36.26	36.89
32	25.81	27.97	31.56	41.85	41.60
64	29.81	34.09	42.69	59.33	57.89
128	36.26	51.90	66.58	77.14	54.89
256	52.65	64.94	62.10	60.84	58.79

FIGURA 26: Erro Quadrático Apresentado pela Rede Neural Sem Wavelets

percentual.

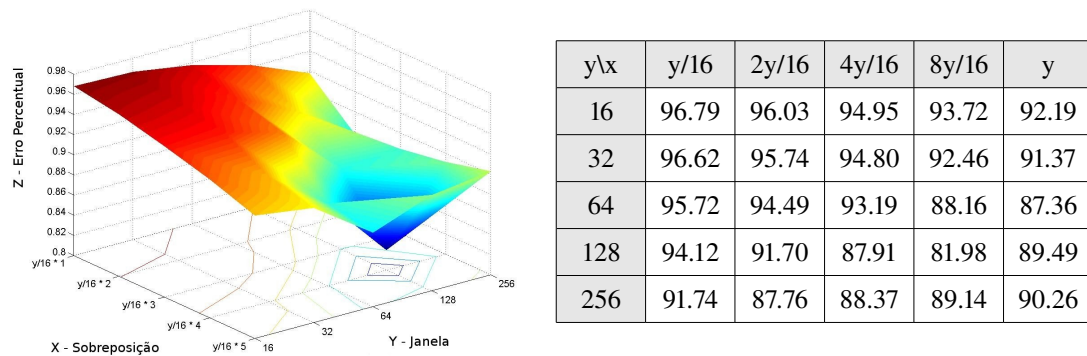


FIGURA 27: Erro Percentual Apresentado pela Rede Neural Sem Wavelets

Na Fig. 28 é apresentado o comportamento em relação ao tempo médio despendido para cada uma das etapas (ET1, ET2, ET3 e ET4).

Na Fig. 29 é apresentada cada uma das quatro etapas isoladamente.

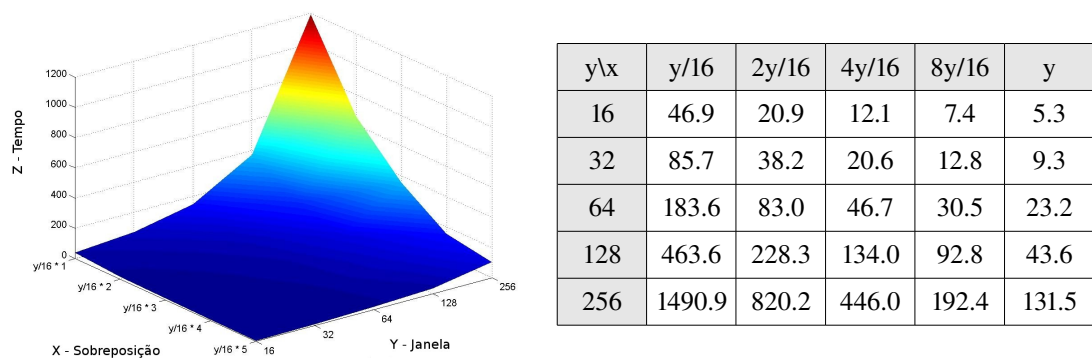


FIGURA 28: Tempo Médio (segundos) para Execução do Experimento

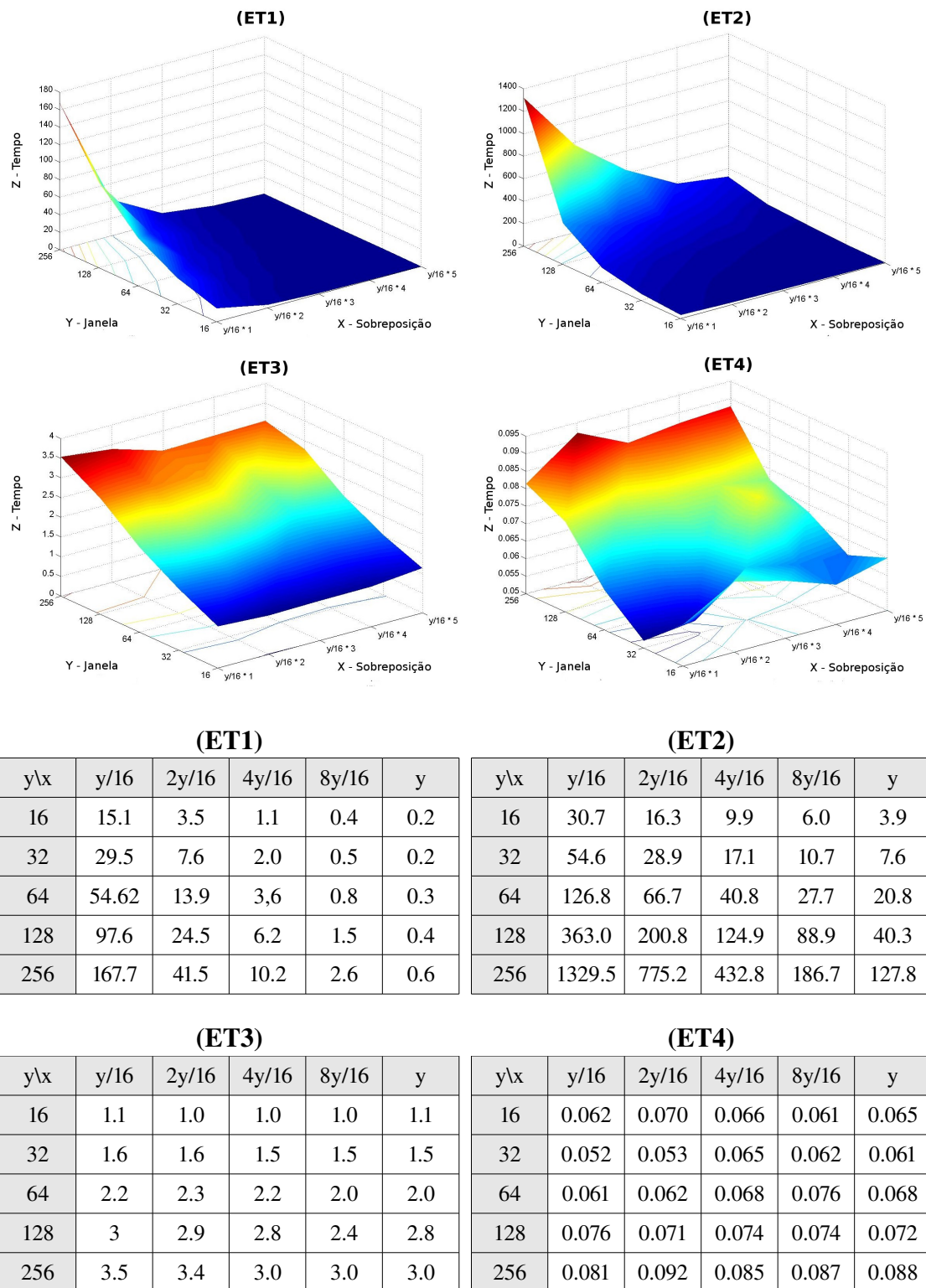
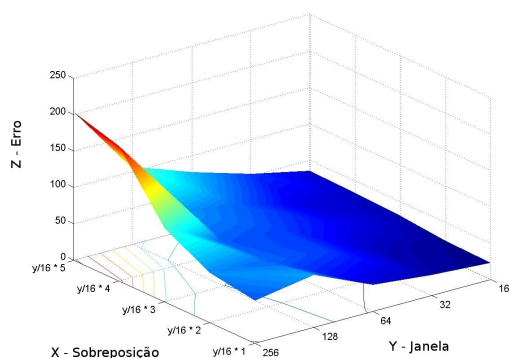


FIGURA 29: Tempo Médio (segundos) para Execução das Etapas (ET1 ... ET4) com Redes Neurais Artificiais

4.2.2 Chroma 6 – Análise de Multiresolução Nível 1

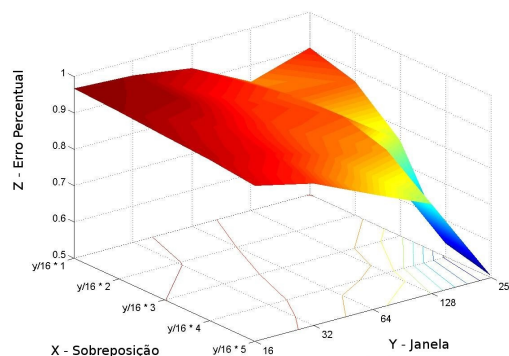
A Fig. 30 apresenta o comportamento padrão do erro quadrático da rede neural para cada combinação proposta, com a aplicação da wavelet Chroma6 aplicando apenas uma decomposição para o pré-processamento dos dados janelados.



$y \backslash x$	$y/16$	$2y/16$	$4y/16$	$8y/16$	y
16	22.76	24.78	30.21	33.21	35.00
32	28.39	30.63	41.39	39.97	52.86
64	36.67	54.13	54.90	61.34	76.92
128	72.46	61.77	60.14	72.46	106.07
256	50.74	68.75	101.50	185.21	203.47

FIGURA 30: Erro Quadrático Médio – Redes Neurais – Wavelet Chma6 – Primeira Decomposição

A Fig. 31 demonstra o comportamento da rede neural para com o acerto percentual.



$y \backslash x$	$y/16$	$2y/16$	$4y/16$	$8y/16$	y
16	96.66	96.00	94.22	94.22	92.78
32	95.79	95.22	91.49	91.49	88.98
64	93.41	89.95	88.26	86.60	81.63
128	85.26	87.00	87.46	83.69	74.51
256	90.26	86.84	76.24	53.67	50.46

FIGURA 31: Acerto Percentual Médio – Redes Neurais – Wavelet Chma6 – Primeira Decomposição

Na Fig. 32 é apresentado o comportamento em relação ao tempo médio despendido para cada uma das etapas (ET1, ET2, ET3 e ET4).

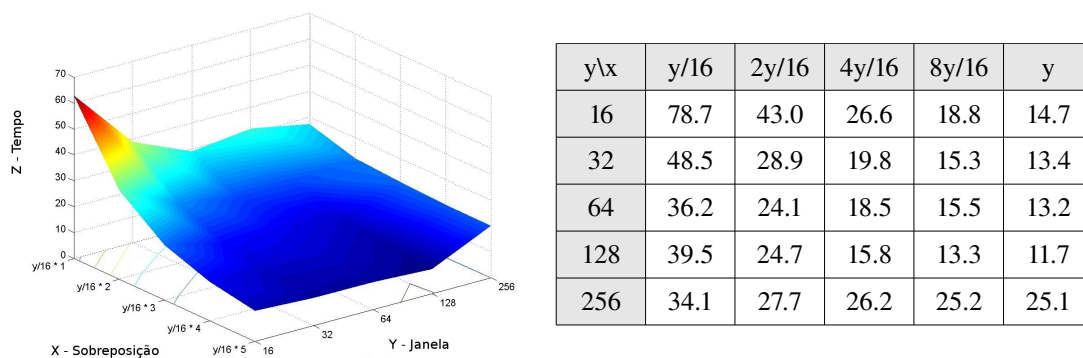
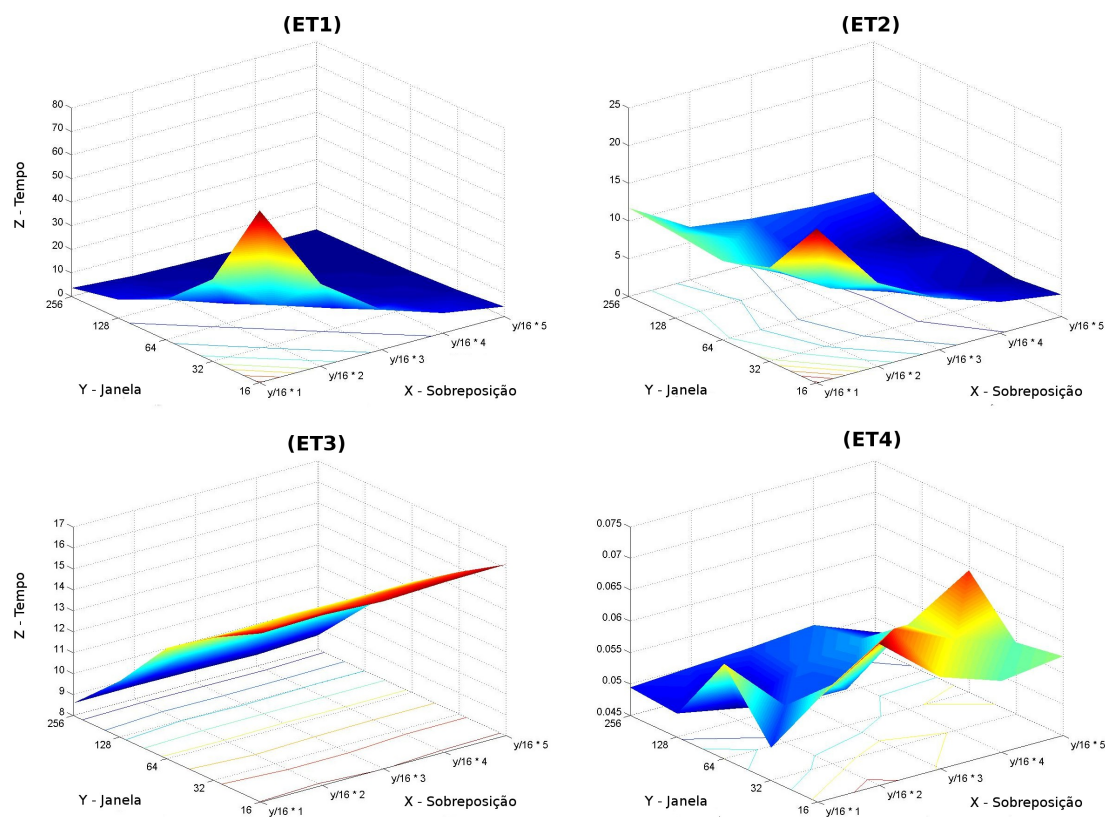


FIGURA 32: Tempo Médio (segundos) para Execução do Experimento

Na Fig. 33 é apresentado cada uma das quatro etapas isoladamente.



(ET1)

y\X	y/16	2y/16	4y/16	8y/16	y
16	45.7	20.8	9.7	4.7	2.3
32	21.7	9.9	4.7	2.3	1.2
64	10.5	4.7	2.3	1.1	0.6
128	4.9	2.2	1.0	0.5	0.3
256	2.1	1.0	0.5	0.3	0.1

(ET2)

y\X	y/16	2y/16	4y/16	8y/16	y
16	23.7	13.0	7.7	4.9	3.2
32	18.0	10.4	6.4	4.4	3.6
64	17.6	11.4	8.1	6.3	4.4
128	27.8	15.8	8.0	6.1	4.6
256	26.3	20.8	20.0	19.3	19.1

(ET3)

y\X	y/16	2y/16	4y/16	8y/16	y
16	9.3	9.1	9.1	9.2	9.2
32	8.7	8.5	8.6	8.5	8.6
64	7.9	7.9	8.0	8.1	8.1
128	6.7	6.7	6.7	6.6	6.8
256	5.6	5.8	5.7	5.6	5.8

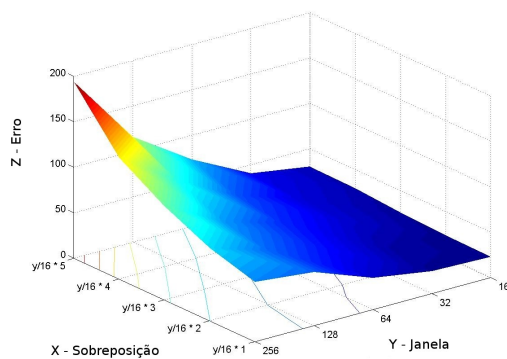
(ET4)

y\X	y/16	2y/16	4y/16	8y/16	y
16	0.064	0.067	0.061	0.062	0.057
32	0.065	0.065	0.059	0.059	0.065
64	0.056	0.059	0.054	0.058	0.051
128	0.065	0.056	0.060	0.062	0.061
256	0.065	0.065	0.064	0.064	0.061

FIGURA 33: Tempo Médio (segundos) para Execução das Etapas (ET1 ... ET4) Redes Neurais Artificiais com Pré-Processamento Wavelet Chma6 – Primeira Decomposição

4.2.3 Chroma 6 – Análise de Multiresolução Nível 2

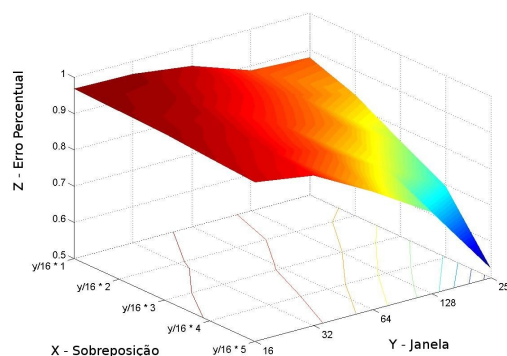
A Fig. 34 apresenta o comportamento padrão do erro quadrático da rede neural para cada combinação proposta, com a aplicação da wavelet Chroma6 aplicando duas decomposições para o pré-processamento dos dados janelados.



$y \backslash x$	$y/16$	$2y/16$	$4y/16$	$8y/16$	y
16	22.80	23.48	25.78	29.39	30.57
32	24.75	28.33	35.28	36.88	40.67
64	34.60	43.81	49.19	56.54	73.41
128	58.80	59.62	68.73	90.33	115.65
256	65.18	80.17	103.21	133.37	193.95

FIGURA 34: Erro Quadrático Médio – Redes Neurais – Wavelet Chma6 – Segunda Decomposição

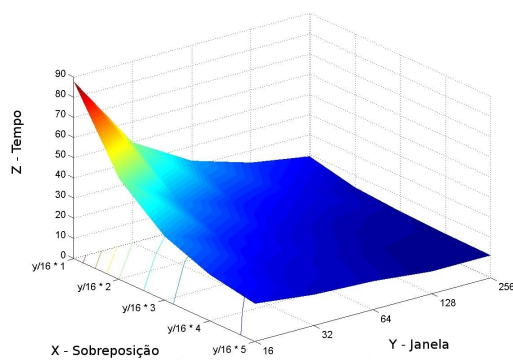
A Fig. 35 demonstra o comportamento da rede neural para com o acerto percentual.



$y \backslash x$	$y/16$	$2y/16$	$4y/16$	$8y/16$	y
16	96.73	96.31	95.84	94.86	93.84
32	96.44	95.33	94.01	92.72	91.35
64	94.33	92.30	89.50	87.71	82.51
128	88.68	87.63	85.11	78.73	72.48
256	87.78	83.38	76.38	69.58	52.60

FIGURA 35: Acerto Percentual Médio – Redes Neurais – Wavelet Chma6 – Segunda Decomposição

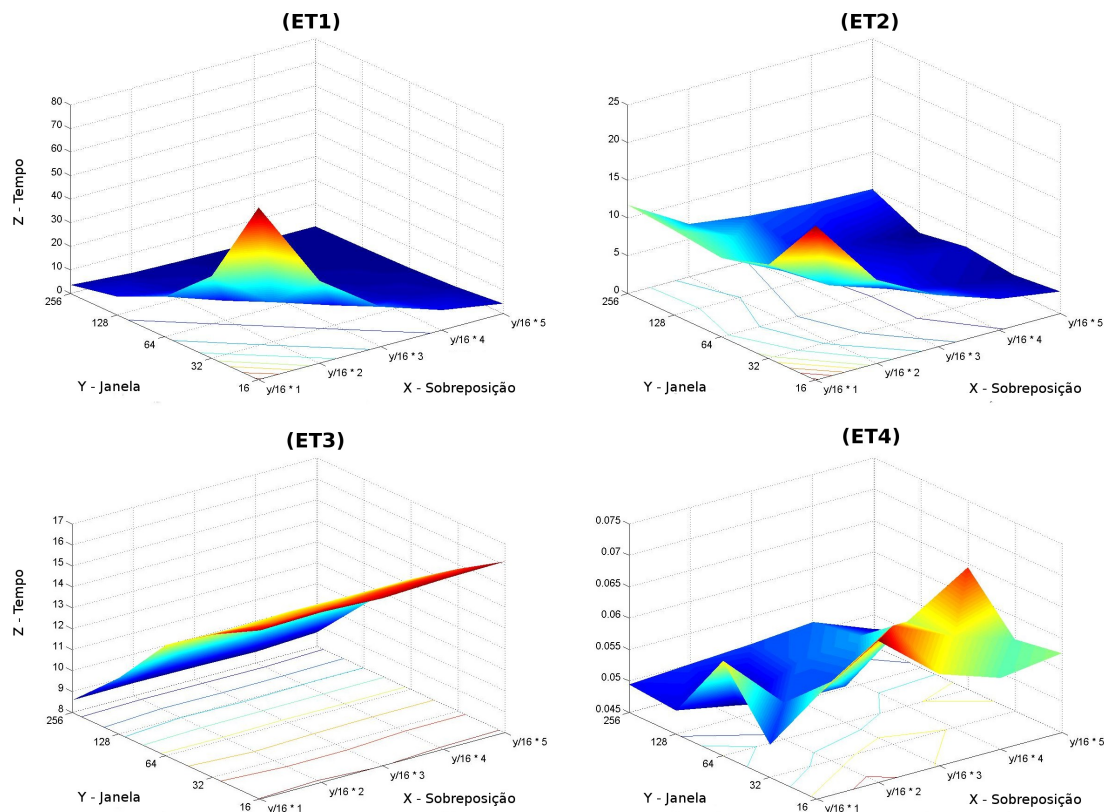
Na Fig. 36 é apresentado o comportamento em relação ao tempo médio despendido para cada uma das etapas (ET1, ET2, ET3 e ET4).



$y \backslash x$	$y/16$	$2y/16$	$4y/16$	$8y/16$	y
16	109.8	62.2	39.7	28.7	23.4
32	62.1	38.8	27.7	22.0	19.2
64	40.7	27.8	21.8	18.8	17.4
128	29.7	23.1	17.3	15.1	13.7
256	23.9	17.4	15.4	14.3	14.0

FIGURA 36: Tempo Médio (segundos) para Execução do Experimento

Na Fig. 37 é apresentado cada uma das quatro etapas isoladamente.

**(ET1)**

$y \backslash x$	$y/16$	$2y/16$	$4y/16$	$8y/16$	y
16	73.2	34.9	16.9	8.4	4.2
32	34.8	16.7	8.3	4.1	2.1
64	16.9	8.1	4.1	2.0	1.0
128	7.9	3.8	1.9	1.0	0.5
256	3.5	1.7	0.9	0.4	0.3

(ET2)

$y \backslash x$	$y/16$	$2y/16$	$4y/16$	$8y/16$	y
16	20.5	11.0	6.8	4.1	2.9
32	12.4	7.5	4.7	3.0	2.3
64	10.4	6.5	4.4	3.5	3.1
128	11.2	8.3	4.4	3.0	2.1
256	11.7	7.0	5.9	5.3	5.0

(ET3)

$y \backslash x$	$y/16$	$2y/16$	$4y/16$	$8y/16$	y
16	16.0	16.1	16.0	16.1	16.2
32	14.8	14.6	14.7	14.8	14.8
64	13.3	13.2	13.3	13.2	13.2
128	10.7	11.0	10.9	11.0	11.0
256	8.6	8.6	8.6	8.5	8.7

(ET4)

$y \backslash x$	$y/16$	$2y/16$	$4y/16$	$8y/16$	y
16	0.061	0.068	0.060	0.056	0.058
32	0.050	0.058	0.064	0.059	0.056
64	0.060	0.051	0.051	0.058	0.064
128	0.049	0.047	0.051	0.053	0.051
256	0.049	0.050	0.051	0.051	0.048

FIGURA 37: Tempo Médio (segundos) para Execução das Etapas (ET1 ... ET4) Redes Neurais Artificiais com Pré-Processamento Wavelet Chma6 – Segunda Decomposição

4.2.4 Chroma 6 – Análise de Multiresolução Nível 3

A Fig. 38 apresenta o comportamento padrão do erro quadrático da rede neural para cada combinação proposta, com a aplicação da wavelet Chroma6 aplicando três decomposições para o pré-processamento dos dados janelados.

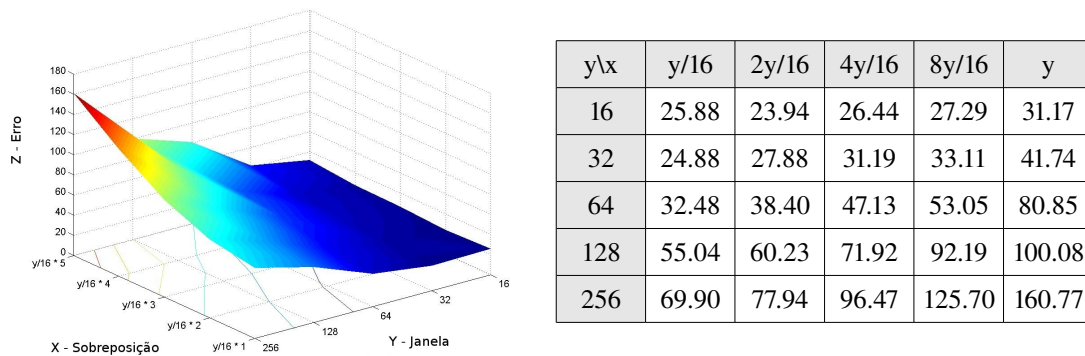


FIGURA 38: Erro Quadrático Médio – Redes Neurais – Wavelet Chma6 – Terceira Decomposição

A Fig. 39 demonstra o comportamento da rede neural para com o acerto percentual.

Na Fig. 40 é apresentado o comportamento em relação ao tempo médio

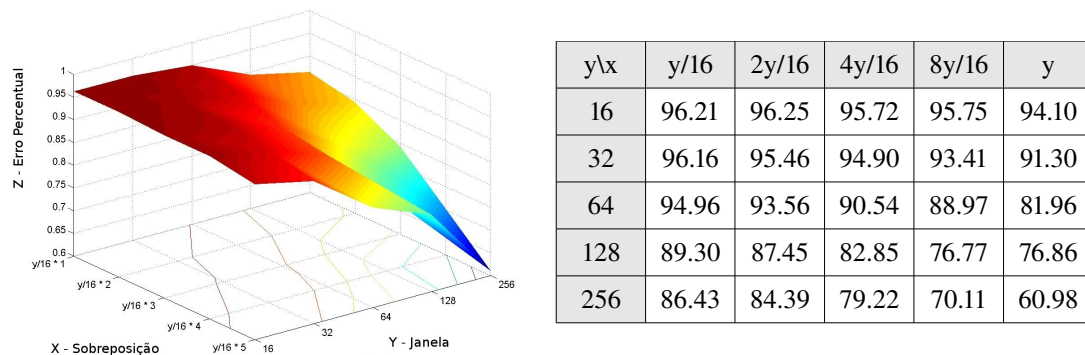


FIGURA 39: Acerto Percentual Médio – Redes Neurais – Wavelet Chma6 – Terceira Decomposição

despendido para cada uma das etapas (ET1, ET2, ET3 e ET4).

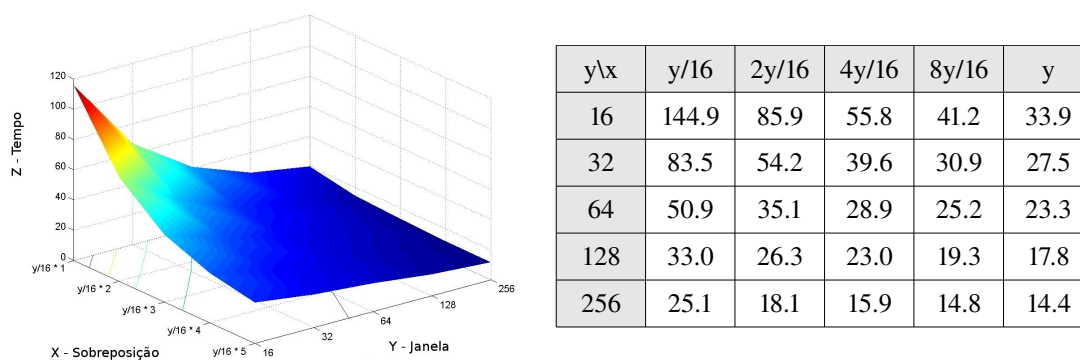
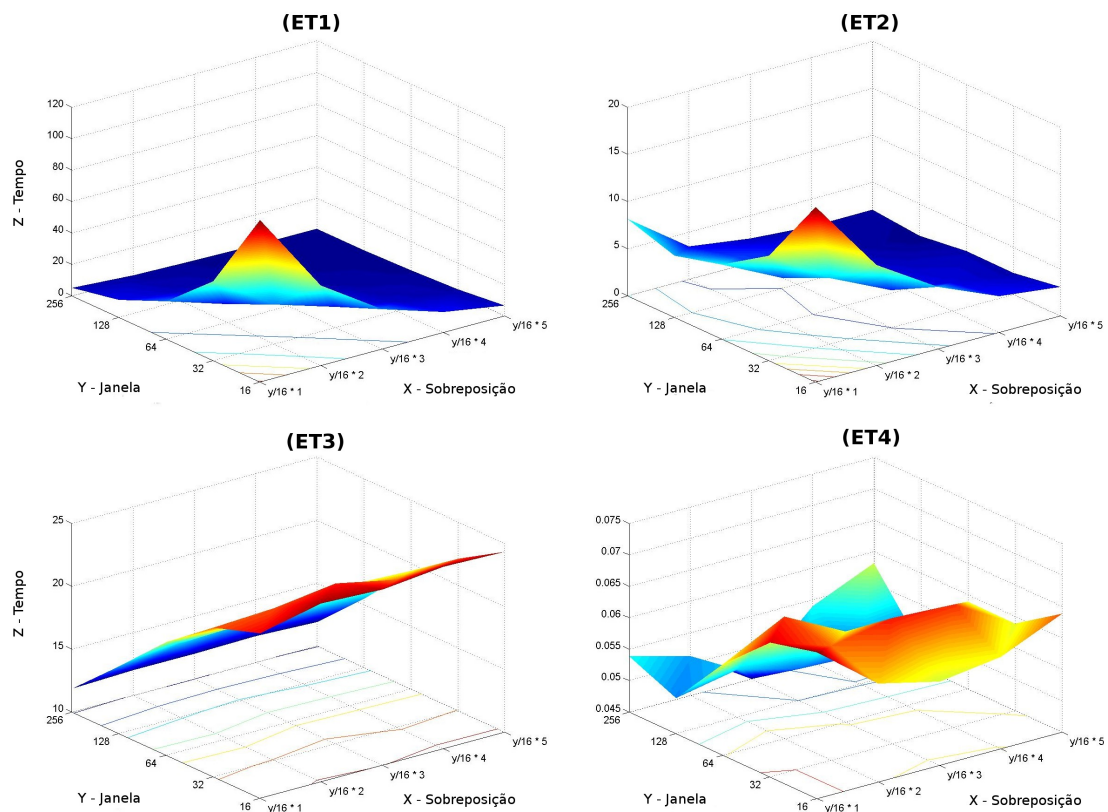


FIGURA 40: Tempo Médio (segundos) para Execução do Experimento

Na Fig. 41 é apresentado cada uma das quatro etapas isoladamente.



(ET1)						(ET2)					
y\X	y/16	2y/16	4y/16	8y/16	y	y\X	y/16	2y/16	4y/16	8y/16	y
16	103.1	50.9	25.2	12.9	6.4	16	18.6	10.7	6.6	3.8	3.1
32	50.5	25.1	12.9	6.1	3.1	32	11.1	7.0	4.0	3.0	2.2
64	23.9	11.7	6.1	3.0	1.6	64	7.9	4.7	3.3	2.6	2.3
128	11.1	5.8	2.9	1.4	0.7	128	6.5	4.7	3.8	2.1	1.6
256	5.0	2.6	1.3	0.7	0.4	256	8.2	3.5	2.5	2.1	2.1

(ET3)						(ET4)					
y\X	y/16	2y/16	4y/16	8y/16	y	y\X	y/16	2y/16	4y/16	8y/16	y
16	23.1	24.2	24.0	24.5	24.4	16	0.068	0.061	0.058	0.060	0.064
32	21.9	22.1	22.7	21.7	22.1	32	0.066	0.063	0.065	0.064	0.059
64	19.0	18.7	19.5	19.6	19.4	64	0.059	0.064	0.059	0.059	0.059
128	15.3	15.8	16.2	15.8	15.42	128	0.051	0.053	0.050	0.050	0.048
256	11.8	12.0	12.0	12.1	11.9	256	0.054	0.051	0.045	0.054	0.058

FIGURA 41: Tempo Médio (segundos) para Execução das Etapas (ET1 ... ET4) Redes Neurais Artificiais com Pré-Processamento Wavelet Chma6 – Terceira Decomposição

4.3 COMPORTAMENTO DO MODELO EM RELAÇÃO AO CONJUNTO DE TREINAMENTO E ÉPOCAS DE TREINAMENTO

Para demonstrar o comportamento generalizado do modelo, foram efetuados alguns testes considerando tamanhos de séries temporais e configurações de épocas diferentes. Para tanto foram consideradas duas situações: janelamento de dezesseis leituras com sobreposição unitária e trinta e duas leituras com sobreposição de duas leituras.

A visualização do comportamento de cada configuração pode ser observada em dois gráficos que são gerados para cada experimento. No primeiro será visto o tempo despendido para uma configuração em específico, e no segundo será demonstrado o crescimento do tempo despendido em relação ao primeiro teste com o mesmo conjunto de dados.

A Fig. 42 mostra o desempenho do modelo utilizando um conjunto de dados de dezesseis leituras com sobreposição unitária.

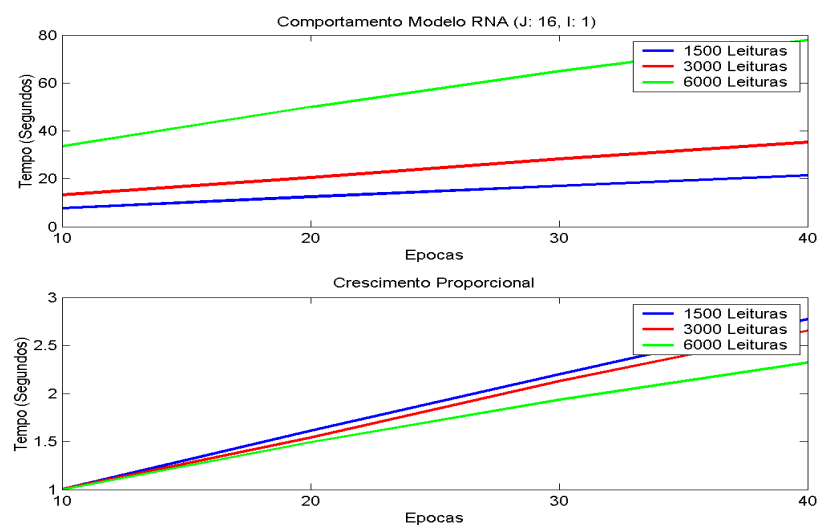


FIGURA 42: Tempo médio para cada configuração do experimento utilizando conjunto de dados de 1500 , 3000 e 6000 leituras, treinados a 10, 20, 30 e 40 épocas, com janelamento de 16 leituras e intercalamento de 1.

A Fig. 43 expõe o desempenho do modelo com a aplicação de wavelets para o pré-processamento dos dados de treinamento.

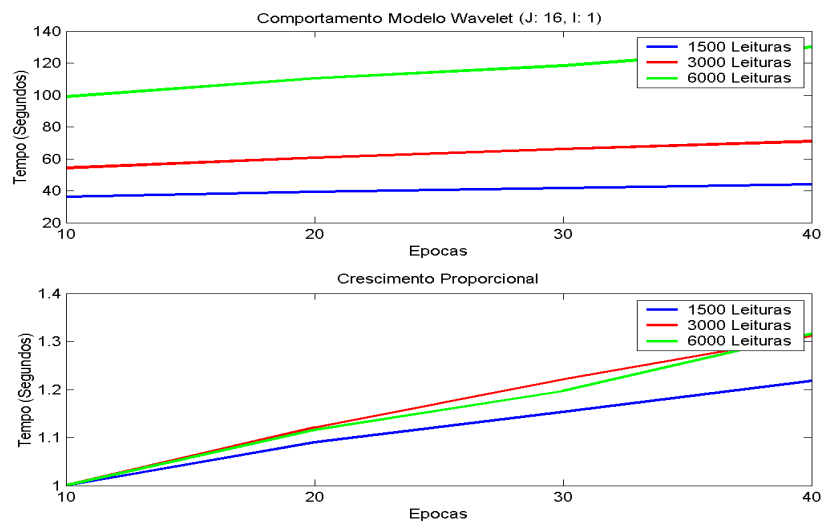


FIGURA 43: Tempo médio (segundos) para cada configuração do experimento utilizando conjunto de dados de 1500 , 3000 e 6000 leituras, treinados a 10, 20, 30 e 40 épocas, com janelamento de 16 leituras e intercalamento de 1 pré-processado por Chroma6.

Nos moldes dos outros gráficos a Fig. 44 e Fig. 45 demonstram o comportamento quando aumentado o janelamento de dados e a sobreposição para treinamento.

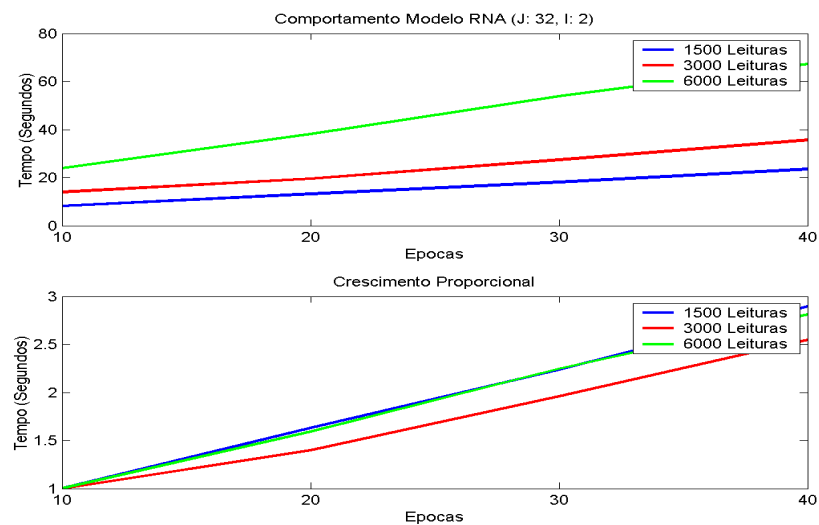


FIGURA 44: Tempo médio para cada configuração do experimento utilizando conjunto de dados de 1500 , 3000 e 6000 leituras, treinados a 10, 20, 30 e 40 épocas, com janelamento de 32 leituras e intercalamento de 2.

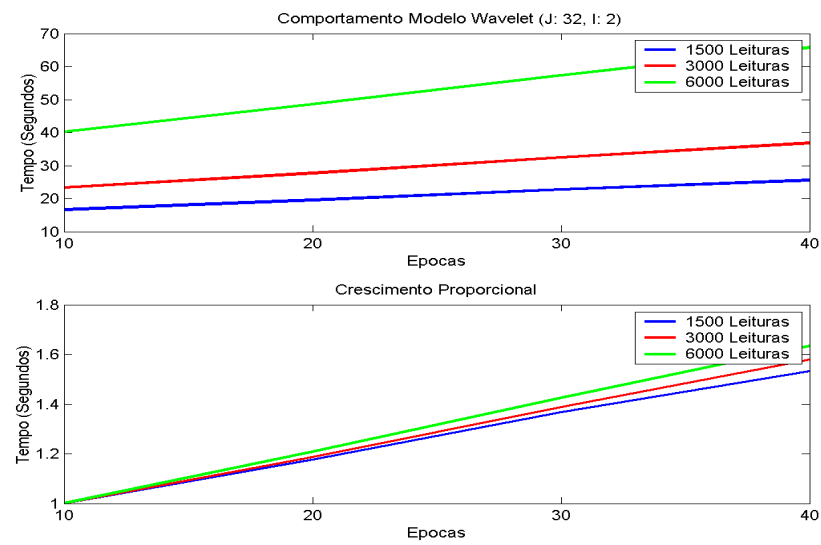


FIGURA 45: Tempo médio (segundos) para cada configuração do experimento utilizando conjunto de dados de 1500 , 3000 e 6000 leituras, treinados a 10, 20, 30 e 40 épocas, com janelamento de 32 leituras e intercalamento de 2 pré-processado por Chroma6.

5 DISCUSSÃO

O primeiro problema envolvido na pesquisa está relacionado a escolha da wavelet que pudesse proporcionar bons resultados, para isto fora montado um projeto para análise de experimento candidatando 20 wavelets, conforme descrito no capítulo de metodologia. A variável analisada para selecionar a onda mais pré-disposta a apresentar bons resultados foi o erro quadrático, o resultado é apresentado na Fig. 46.

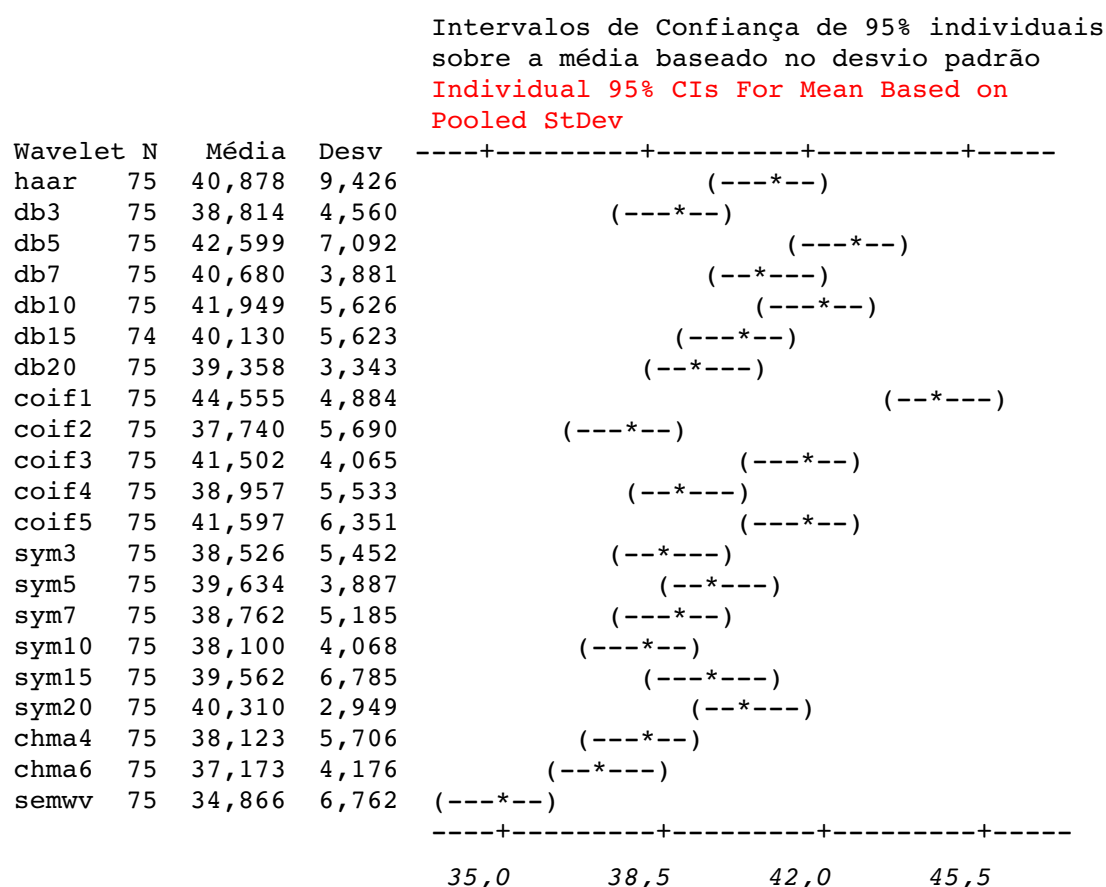


FIGURA 46: Análise de variância efetuada sobre as 75 repetições para cada wavelet.

Fez-se a análise de variância considerando nível de significância de 5%, assim a utilização de redes neurais de perceptrons de múltiplas camadas sem o pré-processamento wavelet é melhor do que no caso onde o pré-processamento ocorre com haar, db3, db5, db7, db10, db15, db20, coif1, coif2, coif3, coif4, coif5, sym3, sym5, sym7, sym10, sym15, sym20 ou chma4. O que não pode ser concluído para a utilização de chma6 no pré-processamento. Indicando esta como candidata mais propensa a apresentar bons resultados em um estudo mais aprofundado.

Pela análise de variância efetuada pode-se constatar que a diferença entre o erro quadrático médio apresentado pelas execuções utilizando wavelet Chroma6 e sem a utilização de nenhum pré-processamento é de 2,3 m³/s. Outro fato a ser considerado é o desvio padrão, que no caso da utilização da wavelet de Chroma6 apresenta um desvio relativamente pequeno em comparação aos outros testes.

Pela análise de variância efetuada, avaliando-se os resultados do experimento foi feita a seleção da wavelet Chroma6 para as outras etapas da pesquisa, entendendo ser a melhor escolha para obter-se bons resultados devido esta apresentar menor erro em relação as outras wavelets.

A segunda etapa da pesquisa envolve a averiguação do comportamento da wavelet selecionada aplicando combinações de janelamento, sobreposições e decomposições pré-estabelecidas.

Sabendo-se que uma atividade de pré-processamento agrega um custo de processamento, aferiu-se também o tempo despendido para a execução de cada etapa do processo. A primeira é justamente a criação do conjunto de treinamento (ET1) e segunda é o treinamento da rede neural (ET2), utilizando o conjunto formulado na primeira etapa. Após treinada a rede, passa-se para a fase de validação onde é criado o conjunto de teste (ET3), com dados não apresentados para a rede na fase de treinamento. Decorrida a formulação do conjunto de teste é feita a inferência deste junto à rede neural (ET4) obtendo assim o retorno do processo preditivo.

A primeira análise que pode ser feita é a de que utilizando wavelets como um pré-processo existem bons resultados. Prevendo que, devido à característica do processo de transformação wavelet, os resultados piorassem (pois com a decomposição é reduzida a quantidade de dados para inferência), o que realmente aconteceu em alguns dos casos, mas atendo-se aos casos onde o resultado foi positivo vê-se que a diferença entre o erro quadrático médio da aplicação ou não de wavelets no pré-processamento é mínimo, onde não menor que a aplicação de pura e simplesmente a rede neural.

Este fato é interessante sobre dois aspectos: primeiramente constata-se que wavelet pode realmente extrair características de séries temporais, uma vez que a utilização dos coeficientes da transformada são em menor número que os dados entrados a esta transformada. O segundo aspecto é a de que a série temporal possui

características a serem extraídas.

Analisando os gráficos apresentados no capítulo de resultado pode-se constatar que as redes neurais aplicadas ao processo preditivo da série temporal de vazão do Rio Itajaí Açu realmente apresentam bons resultados. Destacam-se aqui as execuções onde é empregado um janelamento de 16 leituras com sobreposição unitária e nas execuções onde são agrupadas 32 leituras com sobreposição de duas leituras.

Com a aplicação de uma atividade de pré-processamento, a extração de características, reduz-se o número de parâmetros necessários para o treinamento da rede, obtendo-se como resultado um treinamento mais rápido assim como sua inferência.

Pelos gráficos apresentados no capítulo de resultados, constata-se realmente mudanças pequenas na erro entre a aplicação e não aplicação do pré-processamento wavelet. Principalmente para os casos onde utiliza-se janelamento de trinta e duas leituras com sobreposição de duas e janelamento de dezesseis leituras e sobreposição unitário. A diferença do erro quadrático médio entre elas no primeiro caso fica três centésimos melhor com a utilização da atividade de pré-processamento como é possível verificar comparando as Fig. 26 e Fig. 30, o que não chega a ser percebido em seu uso como constata-se na comparação da Fig. 27 e Fig. 31.

A decomposição para uma segunda resolução mantém seu erro quadrático médio estável, como pode-se averiguar analisando as Fig. 38, Fig. 26 e Fig. 30, reduzindo ainda mais o número de parâmetros para treinamento e inferência.

Considerando o tempo médio gasto para executar as etapas ET1, ET2, ET3 e ET4, as resoluções 2 e 3 crescem de forma a apresentar um tempo quase dobrado.

Se for analisado o tempo despendido para cada etapa pelas figuras, constata-se que o tempo despendido para o pré-processamento passa de 15 segundos para a formulação do conjunto de treinamento sem o pré-processamento para 103 segundos na utilização de wavelets como pré-processador. Mas em contrapartida o tempo gasto pelo treinamento da rede neural pode ser reduzido de 30 para 18 segundos, como constata-se nas Fig. 29 e Fig. 41.

Para o caso onde são coletadas trinta e duas leituras com sobreposição de uma, a combinação rede neural com pré-processamento wavelet de uma decomposição se mostra interessante. Apresentando resultados muito parecidos mas com tempo menor ao

despendido pela utilização de pura e simplesmente da rede neural. Mais uma vez destaca-se que o despendimento maior no processo, pelo qual é aplicado wavelet, é no pré-processamento, assim, em termos quantitativos tem-se que dos 48 segundos 21 são gastos no pré-processamento. A diferença em termos de erro médio quadrático é de 2.58 melhor sem a utilização do processamento wavelet, e em termos de tempo, 37 segundos mais rápido aplicado o pré-processamento.

Em uma segunda decomposição, os resultados têm uma sensível melhora. Contudo o tempo despendido para o pré-processamento aumenta. A segunda decomposição funciona extraíndo coeficientes wavelets dos coeficientes extraídos na primeira decomposição. Desta forma, o número de coeficientes é reduzido conforme aumenta-se o grau de decomposição, reduzindo o número de coeficientes reduz-se a complexidade para o treinamento da rede neural, o que torna esta tarefa mais simples. É possível constatar, pelas Fig. 32, Fig. 36 e Fig. 40 que o tempo de pré-processamento fica a cada decomposição maior e em alguns casos este tempo não é mais compensado pelo tempo economizado no treinamento da rede.

Comparando os resultados entre as diversas configurações, é possível notar que para casos onde o janelamento e a sobreposição são grandes (ex: J 256 S 256, J 256 S 128) o erro apresentado nos casos onde a wavelet fora utilizado é agravado (conforme Fig. , mas nota-se uma tendência de melhora mas muito sensível com o decorrer das decomposições.

Analisando a combinação de janelas pode-se verificar que o fato percebido anteriormente, onde o pré-processamento aumenta e o tempo despendido para treinamento diminui é caracterizado ao delongo do tempo.

Existe um certo equilíbrio entre a utilização da atividade de pré-processamento para as redes e a utilização delas sem o pré-processamento que está vinculado ao tamanho da série que será utilizada para treinamento, épocas que serão utilizadas para o treinamento e o tempo gasto para o mesmo.

O pré-processamento wavelet pode reduzir a longo prazo o tempo gasto para o treinamento, mas em contrapartida necessita de um gasto computacional maior em seu pré-processamento. Como pode-se notar nas Fig. 42, Fig. 43, Fig. 44 e Fig. 45, o crescimento proporcional a sua primeira execução, utilizando apenas redes neurais é

maior que o crescimento proporcional utilizando wavelets.

6 CONCLUSÕES

Os resultados obtidos com base nos experimentos, indicam que as wavelets podem realmente extrair características, apresentando desta forma uma representação compacta (coeficientes de retorno da decomposição) do sinal mantendo bom índice de erro médio quadrático (Fig. 38, Fig. 34 e Fig. 30). Testes indicaram que dentre as candidatas (Haar, Daubechies 3, 5, 7, 10 e 15, Coiflet 1, 2, 3, 4 e 5, Symlet 3, 5, 7, 10, 15 e 20, Chroma 4 e 6), a wavelet Chroma6 se destacou comparado aos erros gerados pela utilização de outras wavelets para efetuar o pré-processamento (Fig. 23). Sobre a análise efetuada pode-se constatar a grande diferença entre aplicações de wavelets diferentes.

O modelo híbrido wavelet/neural foi desenvolvido, apresentando em alguns casos vantagem sobre o modelo puramente neural comparando o tempo para treinamento da rede (fato que se constata na comparação das Fig. 29 e Fig. 33). Este fato deve-se à quantização executada pelas wavelets que reduzem o número de dados necessários para treinamento de redes neurais, minimizando assim o tempo despendido para o treinamento da mesma.

A propriedade de multiresolução das wavelets pode ser utilizado para reduzir o número de dados utilizado para o treinamento das redes, simplificando ainda mais o gasto computacional para o treinamento. É possível notar diferença nos testes realizados com múltiplas decomposições, como é o caso da segunda decomposição (Fig. 34) que apresenta resultados, em alguns casos, média de erro quadrático melhores que os executados em uma primeira decomposição (Fig. 30).

Baseado sobre os testes executados pode-se concluir que alterações na parametrização wavelet/redes neurais artificiais (envolvendo neste ponto tipo da wavelet, tamanho da janela, sobreposição utilizada e tamanho do conjunto de treinamento), pode-se chegar a bons resultados balanceando tempo e erro, indicando desta forma que o modelo possui um ponto de equilíbrio.

6.1 TRABALHOS FUTUROS

Sugere-se a partir deste trabalho futuras pesquisas envolvendo outros sinais de séries temporais, assim como outras wavelets para o pré-processamento dos dados para

a rede neural para verificar a eficiência do modelo. Outro problema de pesquisa está vinculado á alteração da configuração da rede neural quanto ao número de neurônios na camada intermediária, bem como ao algoritmo de aprendizagem vinculado ao treinamento da mesma.

7 REFERÊNCIAS BIBLIOGRÁFICAS

- BOTELLA, F.; ROSA-HERRANZ, J.; GINER, J.J. et al. A real-time earthquake detector with prefiltering by wavelets. Computer & Geosciences, v. 29, no. 7, p. 911-919, 2003.
- CANHOLI, A. P., Drenagem urbana e controle de enchentes. São Paulo: Oficina de Textos, 2005.
- DAUBECHIES, I. Ten Lectures on Wavelets (C B M S - N S F Regional Conference Series in Applied Mathematics), v. 61, Soc for Industrial & Applied Math, Philadelphia, 1992.
- DAUBECHIES, Where do wavelets come from? - A personal point of view. IEEE, v. 84, no. 4, 1996.
- DOVICCHI, J. C. L., Novos Coeficientes Wavelets Baseados em Intervalos Musicais para a Análise de Timbres de Instrumentos Acústicos, Tese apresentada para obtenção de título de Doutor em Engenharia Elétrica, Universidade Federal de Uberlândia, 1999.
- FRANK B.; PINHEIRO A., Enchentes na Bacia do Rio Itajai: 20 anos de experiência. Blumenau: Edifurb, 2003.
- FREEMAN, J. A.; SKAPURA, D. M. Neural networks: algorithms, applications, and programming techniques. Massachusetts: Addison-Wesley, 1991.
- GRAPS A. An Introduction to Wavelets. IEEE Computer Science and Engineering, v. 2, no. 2, 1995.
- GRIFFIN, D. W. et al. Signal Estimation from Modified Short-Time Fourier Transform. IEEE Transactions on Acoustics, Speech and Signal Processing, v. 8, no. 2, p. 236 - 243 1983.
- GOSWAMI, J. C.; CHAN, A. K. Fundamentals of Wavelet - Theory, Algorithms, and Application. New York: Wiley-Interscience, 1999.
- HAAR A. Zur Theorie der orthogonalen Funktionensysteme. Mathematische Annalen 69, p. 331 - 371, 1910.
- HAYKIN, S. Redes Neurais: princípios e prática / Simon Haykin; trad Paulo Martins Engel. - 2.ed. Porto Alegre: Bookmann, 2001.
- IMRIE C. E.; DURUCAN S. River flow prediction using artificial neural networks: generalisation beyond the calibration range. Journal of Hydrology, v. 233, no. 1-4, p. 138 - 153, 2000.
- KALAYCI, T., OZDAMAR, O. Wavelet processing for automated neural-network

- detection of skipe. IEEE Engineering in Medicine and Biology Magazine, v. 14, no. 2, p. 160 - 166, 1995.
- KELLOGG, W. W.; SCHNEIDER, S. H. Climate stabilization: for a better or for worse?. Science, v. 186, no. 4170, p. 1163 - 1172, 1974.
- KHADRA L.; DICKHAUS H. Representation of ECG-late potenciales in the time frequency plane. Journal of Medical Engineering and Technology, v. 17, no. 6, p. 228 - 231, 1993.
- KRÖSE, B. J.; VAN DER SMAGT, P. An Introduction to Neural Networks. Amsterdam: University of Amsterdam, 1996.
- LIN, Z. S.; BIAN W. L.; YOU W. H. The Wavelets and Hierarchies of the Climate System. Meteorology and Atmospheric Physics, v. 61, no. 1-2, p. 19 - 26, 1996.
- LUGER, G. F., Inteligência Artificial: estruturas e estratégias para a solução de problemas complexos; trad. Paulo Engel. - 4.ed. Porto Alegre: Bookmann, 2004.
- MALLAT, S. A Wavelet Tour of Signal Processing - 2.ed. London: Academic Press, 1999.
- MARR, D. Vision. New York: Freeman Publishers, 1982.
- MERTINS, A. Signal Analysis, Wavelets, Filter Banks, Time-Frequency Transform and Application. Chichester: John Wiley & Sons, 1999.
- MICHELI-TZANAKOU, E. Supervised and unsupervised pattern recognition: feature extraction and computational intelligence. Boca Raton: CRC Press, 2000.
- MORETTIN P. A., Ondas e Ondaletas: Da Análise de Fourier à Análise de Ondaletas. São Paulo: EdUSP, 1999.
- NAKKEN, M. Wavelet analysis of rainfall-runoff variability isolating climatic from anthropogenic patterns. Environmental Modelling and Software, v. 14, no. 4, p. 283 - 295, 1999
- PERCIVAL, D. B.; Walden A. T. Wavelet Methods for Time Series Analysis. Cambridge: Cambridge University Press, 2000.
- REZA, A. M., From Fourier Transform to Wavelet Transform Basic Concepts. Spire Lab, UWM, White Paper, 1999.
- ROSENBLATT F. The perception: a probabilistic model for information storage and organization in the brain. Psychological Review, v. 65, no. 6, p. 386 - 408, 1958.
- ROWLEY H. A.; BALUJA S.; KANADE T. Neural Network-Based Face Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, v. 20, no. 1, p. 23 - 38, 1998.

- RUSSELL, S. J.; NORVIG, P. Inteligência Artificial: tradução da segunda edição. Rio de Janeiro: Elsevier - Campus, 2004.
- SENHADJI, L.; CARRAULT G. Comparing wavelet transforms for recognizing cardiac patterns. IEEE Engineering in Medicine and Biology Magazine, v. 14, no. 2, p. 167 - 173, 1995.
- SINGH, R.; VASQUEZ, R. E.; SINGH, R. Comparison of Daubechies, Coiflet, and Symlet for edge detection. Proc. SPIE, v. 3074, p. 151 - 159, 1997.
- SOLTANI, S. On the use of the wavelet decomposition for time series prediction. Neurocomputing, v. 48, no. 1, 2002.
- SOUZA, E. M.; PAGAMISSE, A.; MENEGUETTE Jr, M. et al. Processamento de sinais com wavelets: redução do efeito do multicaminho do sinal GPS-SBMAC:-TEMA - Tendencias em Matematica Aplicada e Computacional, 2005
- TANG, Y. Y.; LIU, J.; YANG, L. H. Wavelet Theory and its Application to Pattern Recognition. Danvers: World Scientific, 2000.
- TUCCI, C. E. M. Regionalização de Vazões. Rio Grande do Sul: Ed. Universidade, 2002.
- TUCCI, C. E. M.; MARENGO J.; DIAS, P. L. S. et al. Previsão de vazões na bacia do Rio São Francisco com base na previsão climática. Relatório técnico ANEEL/OMM/98/00. Porto Alegre: IPH-UFRGS. 385p. 2005.
- UNSER, M.; ALDROUBI A. A Review of Wavelets in Biomedical Application. Proceedings of the IEEE, v. 84, no. 4, p. 626 - 638, 1996.
- VIANELLO, R. L.; ALVES A. R. Meteorologia básica e aplicações. Viçosa: UFV, 2000.
- WELCH, P. D. The Use of Fast Fourier Transform for the Estimation of Power Spectra: A Method Based on Time Averaging Over Short, Modified Periodograms. IEEE Transactions on Audio and Electroacoustics, v. 15, no. 2, p. 70 - 73, 1967.
- WORNELL, G. Signal processing with fractals: a wavelet-based approach. Indianapolis: Prentice-Hall, 1995.
- YNDESTAD H., The influence of the lunar nodal cycle on Arctic climate. ICES Journal of Marine Science, v. 63, no. 3, p. 401 - 420, 2006.

APÊNDICE I

O código dos algoritmos criados para a pesquisa estão apresentados a seguir:

Algoritmo Prever

```
function [net,correto,simulado,tempos] =
prever(JANELA,PROGRESSAO,WAVELET,NIVELDECOMPOSICAO,DADOS,LIMITES,EPOCAS)

[dadosNormalizados,minP,maxP] = premnmx(DADOS);

if (strcmp(WAVELET,['semwv']))
else
    if (strcmp(WAVELET,['chma4']))
        passaBaixa = [0.488877 0.834893 0.218230 -0.127786];
    else
        if (strcmp(WAVELET,['chma6']))
            passaBaixa = [-0.01456106793730538827291720593848
0.46211056360837471368441242702829 0.84245747811804405635882631031564
0.24880228939824242107015095371025 -0.12078962899419119769728325763936
-0.00380607182006964571766016725007];
        else
            passaBaixa = wfilters(WAVELET);
        end
    end
end

TREINAMENTO_INICIO = LIMITES(1);
TREINAMENTO_FIM = LIMITES(2);
TESTE_INICIO = LIMITES(3);
TESTE_FIM = LIMITES(4);

PONTOFINAL_TREINAMENTO = (TREINAMENTO_FIM - TREINAMENTO_INICIO) -
mod( TREINAMENTO_FIM - TREINAMENTO_INICIO ,JANELA);
PONTOFINAL_TREINAMENTO = PONTOFINAL_TREINAMENTO-
mod(PONTOFINAL_TREINAMENTO,PROGRESSAO);

tic;
k=TREINAMENTO_INICIO;
for i=TREINAMENTO_INICIO:PROGRESSAO:TREINAMENTO_INICIO+PONTOFINAL_TREINAMENTO-
JANELA
    if(min([DADOS(i:i+JANELA-1),DADOS(i+JANELA)])>=0)
        if (strcmp(WAVELET,['semwv']))
            wd = dadosNormalizados(i:i+JANELA-1);
        else
            wd =
wavedec(dadosNormalizados(i:i+JANELA-1),NIVELDECOMPOSICAO,passaBaixa,[]);
        end
        dadosTreinamento(:,k-TREINAMENTO_INICIO+1) = wd';
        resultadoTreinamento(k-TREINAMENTO_INICIO+1) =
dadosNormalizados(i+JANELA);
        k = k + 1;
    end
end

tempos(1) = toc;
tic;
```

```

k = 1;
for i=TESTE_INICIO:1:TESTE_FIM-JANELA
    if(min([DADOS(i:i+JANELA-1),DADOS(i+JANELA)])>=0)
        if (strcmp(WAVELET,'semwv'))
            wd = dadosNormalizados(i:i+JANELA-1);
        else
            wd =
wavedec(dadosNormalizados(i:i+JANELA-1),NIVELDECOMPOSICAO,passaBaixa,[]);
        end
        dadosTeste(:,k) = wd';
        resultadoTesteCorreto(k) = dadosNormalizados(i+JANELA);
        k = k+1;
    end
end
tempos(2) = toc;

net=newff(minmax(dadosTreinamento),[10,10,1],{'tansig', 'tansig', 'purelin'},
'TRAINLM');
net.trainParam.epochs = EPOCAS;
net.trainParam.lr = 0.001;
net.trainParam.show = 2;
net.trainParam.goal = 0;

tic;
net = train(net,dadosTreinamento,resultadoTreinamento);
tempos(3) = toc;
resultadoTesteRede = sim(net,dadosTeste);
tempos(4) = toc-tempos(3);

correto = postmnmx(resultadoTesteCorreto,minP,maxP);
simulado = postmnmx(resultadoTesteRede,minP,maxP);

```

Algoritmo TaxaErro

```

function [percentagemVar,erroQuadratico] = taxaErro(correto,simulado)
    INDICEACERTO=0;
    dif = correcto - simulado;
    for i=1:length(dif)
        dif(i)=sqrt(dif(i)^2);
        if (dif(i)<=sqrt(var(cusado)))
            INDICEACERTO = INDICEACERTO+1;
        end
    end
    percentagemVar = INDICEACERTO/length(dif);

    erroQuadratico = mean(dif);

```